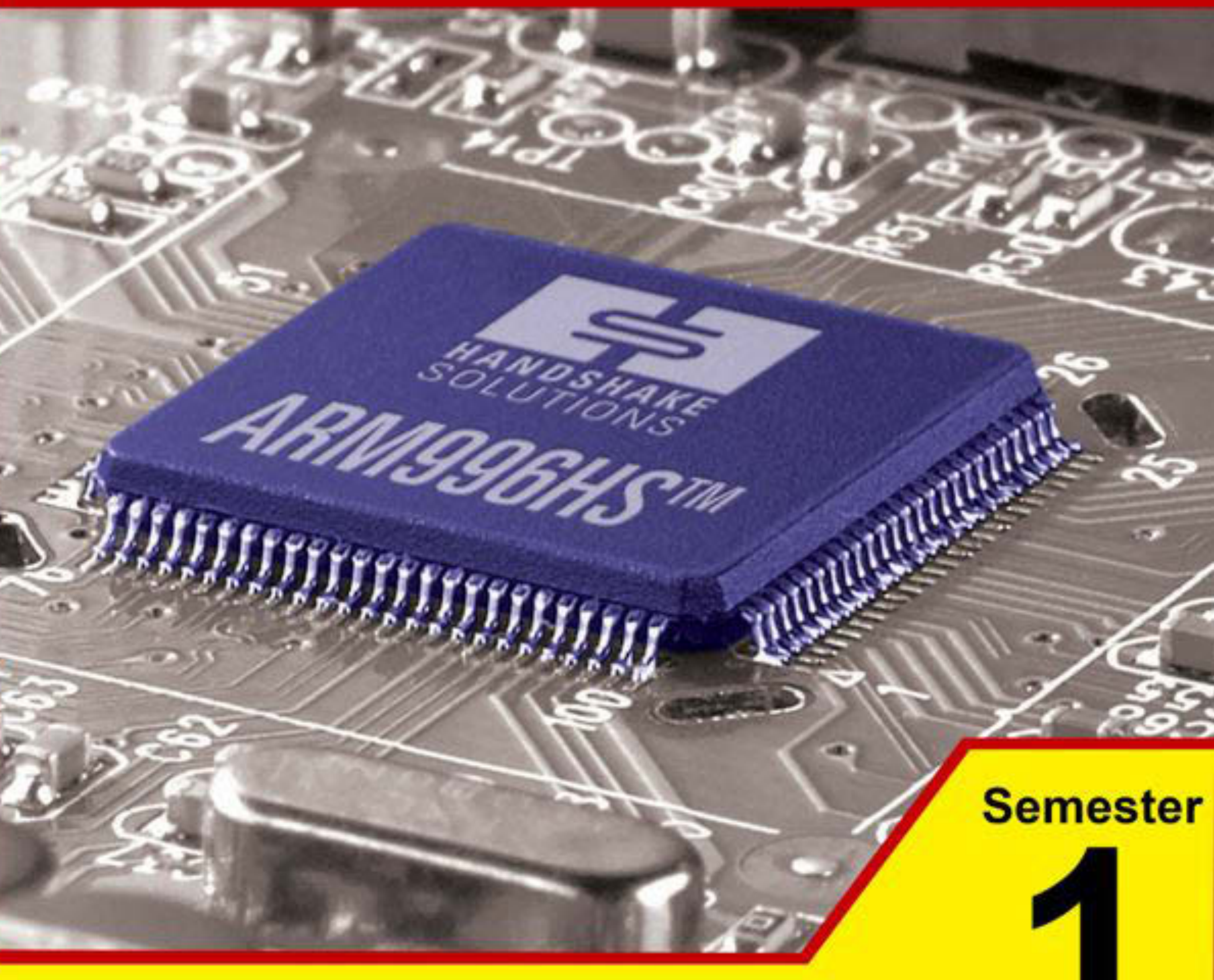
Kementerian Pendidikan Dan Kebudayaan
Republik Indonesia
2013

TEKNIK MIKROPROSESOR



Semester
1

Untuk SMK / MAK Kelas X



Penulis : DJOKO SUGIONO
Editor Materi : WELDAN KHUSUF
Editor Bahasa :
Ilustrasi Sampul :
Desain & Ilustrasi Buku : PPPPTK BOE MALANG

Pusat Pengembangan & Pemberdayaan Pendidik & Tenaga Kependidikan Bidang Otomotif & Elektronika:

Jl. Teluk Mandar, Arjosari Tromol Pos 5, Malang 65102, Telp. (0341) 491239, (0341) 495849, Fax. (0341) 491342, Surel: vedcmalang@vedcmalang.or.id, Laman: www.vedcmalang.com



DISKLAIMER (*DISCLAIMER*)

Penerbit tidak menjamin kebenaran dan keakuratan isi/informasi yang tertulis di dalam buku tek ini. Kebenaran dan keakuratan isi/informasi merupakan tanggung jawab dan wewenang dari penulis.

Penerbit tidak bertanggung jawab dan tidak melayani terhadap semua komentar apapun yang ada didalam buku teks ini. Setiap komentar yang tercantum untuk tujuan perbaikan isi adalah tanggung jawab dari masing-masing penulis.

Setiap kutipan yang ada di dalam buku teks akan dicantumkan sumbernya dan penerbit tidak bertanggung jawab terhadap isi dari kutipan tersebut. Kebenaran keakuratan isi kutipan tetap menjadi tanggung jawab dan hak diberikan pada penulis dan pemilik asli. Penulis bertanggung jawab penuh terhadap setiap perawatan (perbaikan) dalam menyusun informasi dan bahan dalam buku teks ini.

Penerbit tidak bertanggung jawab atas kerugian, kerusakan atau ketidaknyamanan yang disebabkan sebagai akibat dari ketidakjelasan, ketidaktepatan atau kesalahan didalam menyusun makna kalimat didalam buku teks ini.

Kewenangan Penerbit hanya sebatas memindahkan atau menerbitkan mempublikasi, mencetak, memegang dan memproses data sesuai dengan undang-undang yang berkaitan dengan perlindungan data.

Katalog Dalam Terbitan (KDT)

Teknik Elektronika, Edisi Pertama 2013

Kementerian Pendidikan & Kebudayaan

Direktorat Jenderal Peningkatan Mutu Pendidik & Tenaga Kependidikan, th. 2013: Jakarta



KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan yang Maha Esa atas tersusunnya buku teks ini, dengan harapan dapat digunakan sebagai buku teks untuk siswa Sekolah Menengah Kejuruan (SMK) Bidang Studi Keahlian dan Rekayasa, Teknik Elektronika.

Penerapan kurikulum 2013 mengacu pada paradigma belajar kurikulum abad 21 menyebabkan terjadinya perubahan, yakni dari pengajaran (*teaching*) menjadi BELAJAR (*learning*), dari pembelajaran yang berpusat kepada guru (*teachers-centered*) menjadi pembelajaran yang berpusat kepada peserta didik (*student-centered*), dari pembelajaran pasif (*pasive learning*) ke cara belajar peserta didik aktif (*active learning-CBSA*) atau *Student Active Learning-SAL*.

Buku teks ″Teknik Mikroprosesor″ ini disusun berdasarkan tuntutan paradigma pengajaran dan pembelajaran kurikulum 2013 diselaraskan berdasarkan pendekatan model pembelajaran yang sesuai dengan kebutuhan belajar kurikulum abad 21, yaitu pendekatan model pembelajaran berbasis peningkatan keterampilan proses sains.

Penyajian buku teks untuk Mata Pelajaran ″Teknik Mikroprosesor ″ ini disusun dengan tujuan agar supaya peserta didik dapat melakukan proses pencarian pengetahuan berkenaan dengan materi pelajaran melalui berbagai aktivitas proses sains sebagaimana dilakukan oleh para ilmuwan dalam melakukan eksperimen ilmiah (penerapan scientifik), dengan demikian peserta didik diarahkan untuk menemukan sendiri berbagai fakta, membangun konsep, dan nilai-nilai baru secara mandiri.

Kementerian Pendidikan dan Kebudayaan, Direktorat Pembinaan Sekolah Menengah Kejuruan, dan Direktorat Jenderal Peningkatan Mutu Pendidik dan Tenaga Kependidikan menyampaikan terima kasih, sekaligus saran kritik demi kesempurnaan buku teks ini dan penghargaan kepada semua pihak yang telah berperan serta dalam membantu terselesaikannya buku teks siswa untuk Mata Pelajaran Teknik Mikroprosesor kelas X/Semester 1 Sekolah Menengah Kejuruan (SMK).

Jakarta, 12 Desember 2013

Menteri Pendidikan dan Kebudayaan

Prof. Dr. Mohammad Nuh, DEA

DAFTAR ISI

DISKLAIMER (*DISCLAIMER*) .... **ii**

KATA PENGANTAR .... **iii**

DAFTAR ISI .... **iv**

BAB I. PERKEMBANGAN MIKROPROCESSOR **1**

1.1 PENDAHULUAN 1

1.2 Perkembangan Teknologi Mikroprosesor 3

1.3 Perbedaan Antar Mikroprosessor 26

BAB II. KOMPONEN SISTEM MIKROPROSESOR **35**

2.1. DASAR KOMPONEN MIKROPROSESOR 35

2.2. Rangkaian Pembentuk Unit Aritmatik Logik (ALU) 82

BAB III. SISTEM MIKROPROSESOR **110**

3.1. Arsitektur Mikroprosesor Z80 110

3.2. Arsitektur Sistem Mikroprosessor Z80 111

3.3. PIN Hardware Mikroprosessor Z80 123

3.4. Sistem Timing 127

3.5. Fetch Instruksi 128

3.6. Baca – Tulis Memori (R/W) 130

3.7. Siklus Input - Output 131

3.8. Siklus permintaan Bus 132

3.9. Siklus Interupsi Request/Acknowledge 133

3.10. Interupsi Non-Maskable 134

3.11. HALT Exit 135

3.12. Siklus Power-Down Acknowledge 136

3.13. Siklus Power-Down Release 137

3.14. Respon Interupsi 138

3.15. Sistem Hardware 143

3.16. Kecepatan Akses Memori 144

3.17. Interface Memori Dinamik 145



3.18. Pengalamatan Memori 146

3.19. Programmable Pheriperal Interface (PPI) 8255. 162



PERKEMBANGAN SIRKUIT TERPADU MIKROPOSESOR

DESKRIPSI MATERI PEMBELAJARAN

Hingga saat ini mikroprosesor merupakan hasil temuan makin mendominasi pembentukan watak dan karakter manusia dalam mencari solusi permasalahan kehidupan yang semakin komplek, mulai dari keperluan rumah tangga, sistem perbangkan, perkantoran, transportasi dan industri tidak lepas dari pemakaian teknologi mikroprosesor. Untuk memenuhi kriteria sesuai yang diharapkan mikroprosesor mengalami perkembangan yang sangat cepat, sebagai langkah awal agar dapat mengikuti perkembangan mikroprosesor adalah mengenal dan memahami perkembangan revolusi sirkuit terpadu dan perkembangan evolusi mikroprosesor (teknologi semikonduktor).

KOMPETENSI INTI (KI-3)

Kompetensi Dasar (KD):

1. Memahami perkembangan revolusi sirkuit terpadu dan mikroprosesor (teknologi semikonduktor)

Indikator:

1.1. Menjelaskan perkembangan revolusi sirkuit terpadu dan mikroprosesor (teknologi semikonduktor).

1.2. Memahami perkembangan evolusi teknologi mikroprosesor

KOMPETENSI INTI (KI-4)

Kompetensi Dasar (KD):

1. Menjelaskan perkembangan mikropro-sesor

Indikator:

1.1. Menjelaskan perkembangan mikropro-sesor dan interprestasi data hasil pengukuran.

1.2. Menjelaskan perbedaan spesifikasi Mikroprosessor..

KATA KUNCI PENTING

- Sirkuit, terpadu, semikonduktor
- Mikroprosesor, revolusi, evolusi
- Teknologi, spesifikasi



BAB I. PERKEMBANGAN MIKROPROCESSOR

1.1 PENDAHULUAN

Otak manusia merupakan ciptaan Tuhan Yang Maha Kuasa, fungsi otak sangat luas dan salah satunya adalah untuk melakukan perhitungan matematik dan berpikir secara logika. Fungsi tersebut dikenal sebagai pemroses data yang selanjutnya dikenal dengan nama prosesor, sehingga banyak hal dapat dilakukan dengan kedua fungsi mulai dari berhitung, berpikir logika menyimpan informasi dan memanipulasi data. Dengan demikian sudah seharusnya kita bersyukur atas karunia yang diberikan Tuhan YME dalam bentuk otak yang bekerja 24 jam secara terus menerus.

Selain otak ternyata tubuh manusia dilengkapi dengan berjuta sensor yang terpasang di seluruh tubuh, sebagai contoh dengan menggunakan mata kita bisa melihat artinya ada sesuatu di luar tubuh yang data atau informasinya bisa ditangkap mata kemudian dikirim ke otak. Suara yang ada disekitar manusia dapat didengar bahkan data yang diterima otak diproses pada akhirnya berbagai suara yang tersensor oleh telinga dapat dibedakan, apakah suara tersebut suara manusia, suara hewan atau suaran benda jatuh dan sebagainya. Demikian juga sensor yang terpasang pada seluruh permukaan kulit, pada setiap titik permukaan kulit dapat merasakan manakala sedang dicubit, terkena duri pada telapak kaki, digigit nyamuk bahkan tatkala tergores benda tajampun akan terasa. Jadi sensor merupakan jalur masukan data untuk proses diotak, dan dengan adanya sensor yang terpasang pada seluruh tubuh ternyata dapat mengambil data atau informasi di luar tubuh melalui sensor, dan dalam tubuh terdapat jalur data sebagai pengangkut informasi untuk dibawa ke otak untuk diproses.

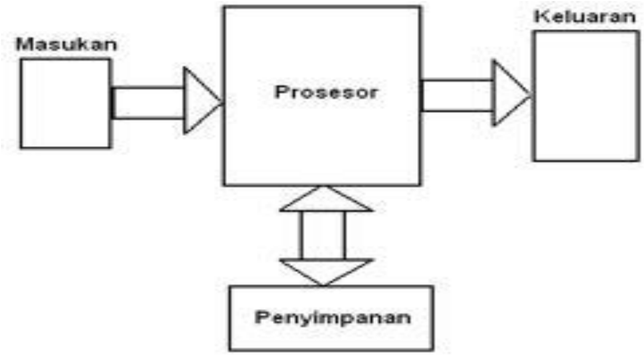
Sebagai hasil atau keluaran proses otak adanya perintah otak ke bagian (organ) untuk melakukan gerakan, sepasang kaki diperintahkan untuk saling bekerjasama melakukan jalan atau lari, jongkok dan melompat. Tangan dapat

diperintah otak untuk memukul, memegang, menulis atau untuk pekerjaan lainnya. Mulut dapat diperintah untuk makan, mengunya, mengigit dengan gigi tertentu dan sebagainya. Jadi anggota tubuh tersebut sebagai aksi keluaran perintah otak, dan merupakan alat yang diperintahkan untuk melakukan sesuatu sesuai dengan kebutuhan.

Identik dengan otak manusia adalah prosesor yang sering disebut sebagai otak dan pusat pengendali sistem komputer, prosesor ini dapat melakukan operasi berhitung, berpikir logika menyimpan informasi dan memanipulasi data seperti yang dilakukan otak manusia. Untuk dapat beroperasi sistem komputer didukung oleh sejumlah komponen di dalamnya, ditinjau secara perangkat keras sebuah prosesor adalah sebuah IC (*chip*) yang mengontrol keseluruhan jalannya sebuah sistem komputer dan digunakan sebagai pusat atau otak dari komputer. Prosesor lebih dikenal dengan istilah "*Microprosessor*", saat ini memiliki kecepatan proses yang luar biasa yaitu mencapai Gigahertz (GHz). Performa kinerja sebuah mikroprosesor sangat ditentukan oleh kecepatan proses dalam mengolah data atau informasi, sebagai penentu kecepatan adalah *clock* yang dipasangkan pada mikroprosesor tersebut.

Dalam sistem komputer sebuah posesor memerlukan data untuk diolah, untuk keperluan tersebut prosesor dilengkapi dengan sistem interface dan melalui sebuah interface inilah data atau informasi dapat dimasukan atau dikeluarkan. Masukan pada sistem komputer diantaranya papan ketik (keyboard), mouse, scanner, pen, sistem sensor, touchscreen dan sebagainya, adapun keluaran sistem komputer diantaranya printer, monitor, LCD, aktuator robot dsb.

Gambar 1.1 menunjukan blok diagram sistem prosesor, yang terdiri dari masukan (*input*), pusat pengolah data (*processor*) dan keluaran (*output*).



Gambar 1.1. Blok diagram minimal sistem prosesor

Dalam minimal sistem prosesor penyimpanan (*storage*) digunakan untuk menyimpan data, hubungan dengan prosesor adalah dua arah karena data bisa dituliskan artinya disimpan dan data bisa dibaca kembali oleh prosesor. Sedangkan masukan data hanya bisa dibaca (satu arah), dan keluaran data hanya bisa dituliskan juga satu arah.

Saat ini sangat banyak merk prosesor yang diperdagangkan dipasaran dunia, namun demikian secara prinsip kerja adalah sama. Perbedaan antara satu prosesor dengan prosesor lainya adalah pada fitur, fasilitas yang dimiliki, kecepatan clock, kapasistas memory dan I/O akse. Merk tersebut diantaranya Asus, Zilog, AMD, Apple, Cyrix, IBM, IDT, Intel dan masih banyak lagi lainya.

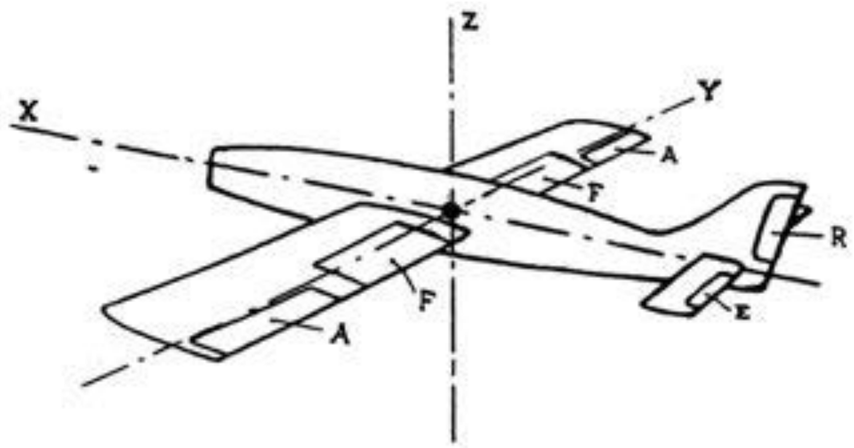
1.2 PERKEMBANGAN TEKNOLOGI MIKROPROSESOR

Dewasa ini penggunaan mikroprosesor sudah menjadi kebutuhan di berbagai bidang baik elektronik, elektrik, permesinan, pengerjaan logam, kayu maupun di dunia otomotip, mengapa demikian karena mikroprosesor merupakan jantung sistem kendali otomasi yang aplikasinya sangat luas bentuk fisiknya sebuah chip (IC) yang sistem kerjanya melalui program. Adapun fungsi mikroprosesor adalah sebagai pengontrol atau pengolah utama dalam suatu rangkaian elektronik yang sering dikenal dengan istilah CPU (Central Processing Unit).

Dalam aplikasinya kontrol menggunakan mikroprosessor sangat luas mulai dari motor step, kontrol PWM, Wattmeter, konverter thyristor, kontrol proses, kontrol terkait dengan berbagai gerakan mekanik, sistem kontrol mesin

otomotip, penerapan hukum kontrol (PID), sistem kontrol pada peralatan modern, kontrol jarak jauh (remote), kontrol otomasi di industri, sistem robot dan peralatan pada otomasi kantor dengan sistem jaringannya.

Sebagai contoh aplikasi mikroprosessor adalah penerapan gerakan dinamis rool dari pesawat terbang dan implementasi kontrol nonlinier yang mengutamakan stabilitas tinggi, akurat dan memiliki respon kecepatan tinggi. Dimana di bidang pesawat udara yang merupakan bodi kaku di angkasa memiliki 6 (enam) derajat kebebasan, sehingga mampu bergerak sepanjang 3(tiga) sumbu x, y, z dan mampu bergerak untuk berputar (rotasi) dalam sumbu yang sama. Untuk itu memerlukan pengaturan bagian-bagian seperti gambar berikut:



Gambar 1.2. Aplikasi Mikroprosessor Pada Pesawat Terbang.

(A: Airlerons, E: Elevator, R: Rudder, F : Flap)

(N.K. Shinha, 1986)

Sebuah mikroprosesor minimal terdiri dari rangkaian digital, register, pengolah logika aritmatika, rangkaian sekuensial yang bekerja berdasarkan algorithma program dalam bentuk kode-kode biner atau bahasa mesin yang telah dimasukkan terlebih dahulu ke dalam sebuah memori.

1. Teknologi Pertama Mikroprosesor.

Prosesor pertama kali dibangun di Amerika serikat yaitu di University of Pennsylvania pada tahun 1946, saat itu sistem yang dikembangkan sudah mengacu pada sistem yang ada saat ini dan dikenal dengan sebutan ENIAC yang merupakan singkatan dari Electronics Numerical Integrator and Calculator. Berat keseluruhan lebih dari 30 ton, disusun dari 17.000 tabung hampa elektron, menggunakan kabel konektor mencapai 500 mil panjangnya, akan tetapi dalam menjalankan instruksi baru mampu sekitar 100.000 operasi per detiknya dan program dilakukan melalui pengaturan jalur kabel koneksi rangkaiannya.

ENIAC dirancang dengan tujuan untuk menghitung tabel tembak artileri (artillery firing table) untuk keperluan tentara Amerika saat perang dunia kedua dengan biaya $500,000, lokasi pengembangan bertempat di United States Army's Ballistic Research Laboratory. Kecepatan ENIAC mencapai seribu kali lebih cepat dibanding mesin elektro-mekanik, dan saat dipublikasikan dikenal dengan istilah otak raksasa ("Giant Brain") yang diusulkan sebagai dasar arsitektur komputer dalam sistem pemeblajaran



Gambar 1.3. Tabung hampa yang digunakan dalam ENIAC



Gambar 1.4. Tabung hampa pasca ENIAC sampai saat ini

Jumlah tabung hampa yang dipasang ENIAC mencapai 17,468, 7,200 diode kristal, 1,500 relay, 70,000 resistor, 10,000 kapacitor dan titik solder mencapai 5 titik solder manual. ukuran total 2.4 m × 0.9 m × 30 m, dengan daya listrik 150 kW. Input dibuat oleh IBM dalam bentuk card reader, dan IBM card punch sebagai pencatat output.

Arsitektur ENIAC menggunakan 10 ring counter yang berfungsi untuk menyimpan digit dan setiap digit menggunakan sejumlah 36 tabung hampa, 10 tabung merupakan tabung dual triode yang membentuk flip-flop dari ring counter. Arithmatika dilakukan dengan menghitung pulsa melalui ring counter dan terjadi carry pulse jika pada counter terjadi "wrapped around", ide dasarnya adalah untuk emulasi roda digit yang digunakan pada mesin mekanik ke dalam bentuk operasi elektronik. ENIAC dilengkapi dengan 20 buah akumuluator yang mampu menampilkan10 digit dengan komplemen 10, sehingga mampu melakukan 5000 operasi penjumlahan dan pengurangan dalam waktu 1 detik.

Untuk melakukan operasi double precision dilakukan dengan menyambungkan satu akumulator dengan akumulator lainnya guna mendukung operasi arithmatik, untuk operasi tersebut ENIAC menggunakan 4 akumulator yang dikendalikan unit khusus *Multiplier* dan mampu melakukan 385 operasi perkalian dalam waktu 1 detik. ENIAC menggunakan 5 akumulator yang dikendalikan unit khusus *Divider/Square-Rooter* untuk melakukan 40 operasi pembagian atau 3 operasi akar perdetiknya.

9 unit ENIAC lainnya difungsikan sebagai unit inisiasi seperti star/stop mesin, unit *Cycling* (used berfungsi untuk sinkronisasi dengan unit lainnya, dan *Master Programmer* berfungsi untuk mengendalikan sekuensial "loop", *Reader* yang dikendalikan oleh IBM punched card reader, *Printer* yang juga dikendalikan oleh IBM punched card punch, *Constant Transmitter*, dan 3 buah tabel fungsi.

Inventor ENIAC adalah John Mauchly dan J. Presper Eckert, sebelum ENIAC secara penuh beroperasi pada Agustus tahun 1944 diusulkan membangun ***E****lectronic* ***D****iscrete* ***V****ariable* ***A****utomatic* ***C****ompute* (EDVAC).



Gambar 1.5. EDVAC (wikipedia)



Gambar 1.6. ENIAC (wikipedia)

EDVAC merupakan binary serial computer yang dapat melakukan operasi penjumlahan pengurangan, perkalian, pembagian dan pengujian otomatis, dilengkapi sistem penyim-panan data ultrasonic serial memory dengan kapasitas 1,000 44-bit word yang selanjutnya dikenal dengan istilah 1,024 word atau sama dengan 5.5 kilobytes.
Adapun komponen sebagai bagian yang tergabung secara fisik dalam EDVAC meliputi:

- *magnetic tape reader-recorder* saat itu dikenal dengan istilah *wire recorder.*
- unit pengendali (*control unit*) dilengkapi dengan sebuah osiloskop.
- unit *dispatcher* yang berfungsi sebagai penerima instruksi dari kontrol dan memori, juga berfungsi sebagai pengarah ke unit lain.



- unit komputasi berfungsi sebagai operasional arithmatik dari pasangan bilangan dalam waktu tertentu dan mengirimkan hasilnya ke memori setelah pengujian dalam unit duplikasi (*duplicate unit*).
- sebuah *timer*
- unit *dual memory* yang terdiri dari 2(dua) set dengan masing-masing terdiri dari 64 mercury akustik delay lines yang berkapasitas 8 word untuk setiap line.

Kecepatan waktu proses pada EDVAC berkisar 864 mikro detik dan untuk proses perkalian dibutuhkan waktu 2900 mikro detik (2.9 mili detik). Prosesor tersebut dibangun dari 6,000 tabung vakum dan 12,000 diode, dibutuhkan daya listrik sebesarc 56 kW. Luasan area lantai yang digunakan EDVAC 490 ft^2 (45.5 m^2) dan beratnya mencapai 17,300 lb (7,850 kg). Sedangkan untuk pengoperasiannya dibutuhkan 30 (tiga puluh) orang, yang digilir untuk setiap 8 jam kerja

2. Teknologi IC Mikroprosesor

Dua tahun kemudian (1948) dengan munculnya perkembangan transistor sebagai pengganti tabung elektron hampa, laboratorium Bell di Amerika serikat membangun prosesor dari bahan transistor. Selanjutnya pada tahun 1958 Texas Instrument membangun rangkaian semikonduktor dalam bentuk IC (Integrated Circuit), hal ini merupakan rangkaian semikonduktor pertama menggunakan chip IC yang dikembangkan Jack Kilby dari USA.

Penemuan IC ini mendorong pengembangan IC Digital (1960), dan mikroprosesor pertama oleh Intel (1971) yaitu mikroprosesor Intel 4004 yang merupakan prosesor 4-bit, bahkan kebanyakan Kalkulator yang digunakan saat ini masih berbasis mikroprosesor 4-bit. Pada tahun yang sama Intel mengeluarkan mikroprosesor 8-bit yang diberi seri Intel 8008, dan di tahun 1973 Intel memperkenalkan mikroprosesor 8-bit modern pertama Intel 8080 yang kecepatan operasinya 10x lebih cepat dari 8008, pengembangan lain diikuti Motorola MC6800.

Pada tahun 1977 Intel memperkenalkan mikroprosesor 8-bit yang diberi merek dagang Intel 8085, mikroprosessor ini dibuat Intel dengan frekuensi clock dengan kecepatan lebih tinggi. Di sisi lain Zilog Corporation membuat mikroprosesor tandingan dari Intel 8085 yaitu dengan dikeluarkannya mikroprosesor Z80. Ditinjau dari sisi spesifikasi, mikroprosesor yang diproduksi Zilog yaitu Z80 memiliki spesifikasi mikroprosesor 8-bit CPU dengan kode obyek kompatibel dengan produksi Intel 8080 CPU. Pada mikroprosesor Z80 dilengkapi keseluruhan register yang ada di Intel 8080 begitu juga instruksinya, dan mendukung keberadaan interupsi di 8080. Selain dari hal tersebut Z80 memiliki kelebihan, meliputi:

- 80 buah instruksi baru, termasuk transfer blok, instruksi manipulasi bit dan string,
- 2 buah index register baru dan duplikat register dan register status.
- tipe interupsi baru untuk Z80 dan devais periperal non-Z80.
- catu daya tunggal +5V.

Beberapa perusahaan di Eropa telah memproduksi Romania dan negara Soviet, komputer ayang diproduksi dengan menggunakan prosessor Z80 diantaranya Radio Shack TRS-80 Model 1 - 4, Sinclair ZX81, Commodore 128D, Franklin Ace 1200, Osborne 1, KayPro I, KayPro II.

Pengembangan berikutnya pada tahun 1978 oleh Intel yaitu dengan diproduksinya generasi mikroprosesor 16-bit yang diberi merk dagang Intel 8086, dan penyempurnaan terus dilakukan oleh Intel sehingga setahun kemudian mengeluarkan prosesor tipe Intel 8088 yang ditinjau dari kecepatan proses dan pengembangan ukuran memori yang jauh lebih besar (expand memory) dibandingkan tipe Intel 8085. Pengelolaan memori sebagai tuntutan pengembangan perangkat software oleh Intel digunakan sistem cache memori, sistem ini merupakan sistem antrian yang mengatur secara sekuensial pemberian instruksi sebelum instrikusi tersebut dijalankan oleh prosesor. Sehingga Intel 8086/8088 dikenal dengan istilah CISC (Complex Instruction Set Computer), sebutan ini diberikan disebabkan banyaknya jumlah dan kompleksitas instruksi yang dapat dioperasikan.



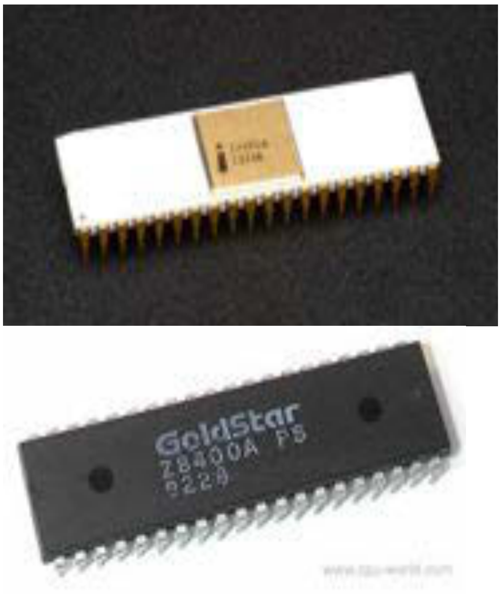
Babak baru pengembangan program aplikasi untuk mendukung mikroprosesor 8088 dilakukan oleh IBM pada tahun 1981 yaitu digunakannya IBM PC sebagai standar komputer dilengkapi dengan program aplikasi seperti spreadsheet dan pengolah kata. Pada tahun 1983 Intel mengeluarkan mikroprosesor 16-bit 80286 yang mampu mengendalikan sistem memori sampai 16 MB, dan tahun 1986 Intel mengeluarkan mikroprosesor 32-bit pertama 80386 yang mampu mengendalikan sistem memori sampai 4 GB. Dilanjutkan pada tahun 1989 Intel mengeluarkan mikroprosesor 32-bit 80486 yang dilengkapi kemampuan memori sampai 4 GB ditambah dengan 8K Cache. Tahun 1993 Intel memperkenalkan mikroprosesor 32-bit yang dikenal dengan istilah Pentium I, dan tahun 1997 diproduksi Pentium II, secara berturut-turut diikuti produksi berikutnya yaitu Pentium III dan Pentium IV dipasarkan keseluruh penjuru dunia pada tahun 2000, pada era inilah mulai dikenal sistem memori dengan teknologi memori RAMBUS menggantikan teknologi SDRAM.

3. Arsitektur Dasar Zilog dan Intel

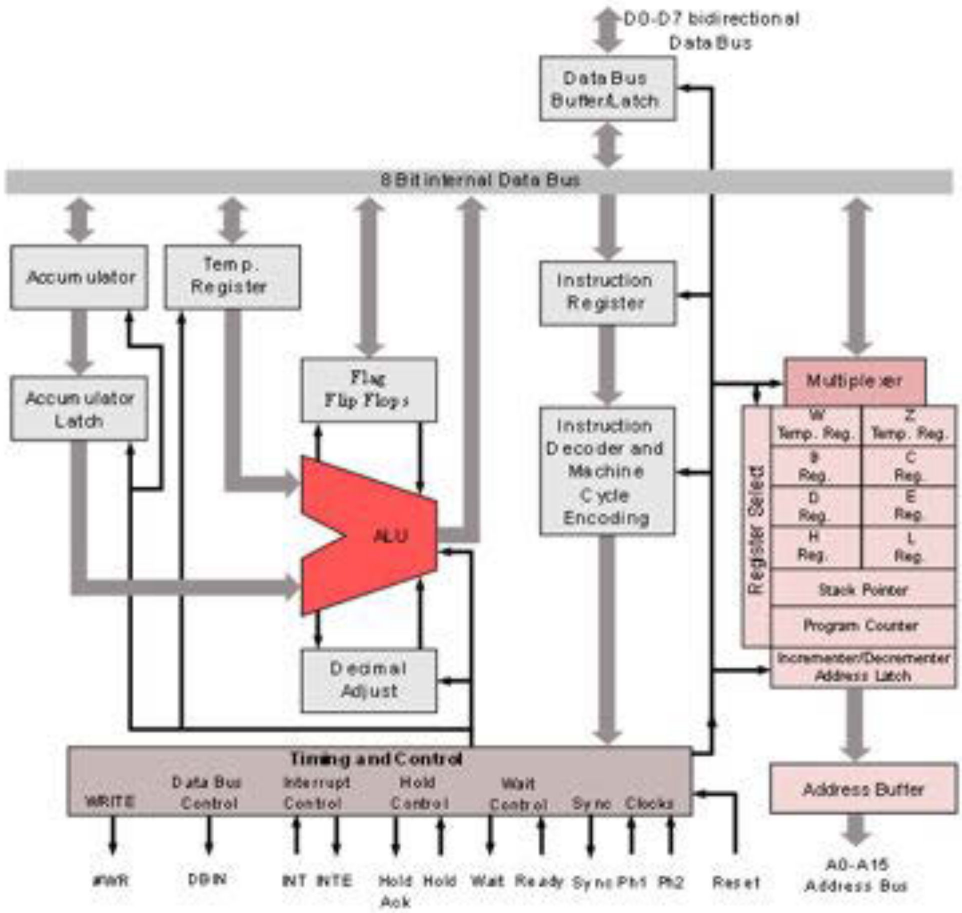
Z80 merupakan prosesor 8-bit mikroprosesor yang dirilis bulan Juli tahun 1976 dengan kecepatan clock 2,5 MHz, pada saat itu Z80 lebih berkembang dibanding Intel 8080 begitu juga dibanding Intel 8085. Secara arsitektur Z80 memiliki kesamaan dalam jumlah bit dengan Intel 8080 yaitu untuk jalur data menggunakan 8-bit dan jalur penunjukan alamat baik port I/O maupun alamat memori menggunakan 16-bit. Z80 dapat menjalankan semua op-code Intel 8080 termasuk 80 kode barunya yang meliputi operasi instruksi 1, 4, 8 dan 16-bit dan juga transfer blok dan instruksi blok I/O.

Kelompok register yang digunakan dalam Z80 terdiri dari 2(dua) blok, yang dibagi menjadi 2(dua) bank register termasuk register A dan register F yang dapat saling ditukarkan isi datanya. Dengan demikian sangt memungkinkan penggunaan sistem operasi dengan kecepatan tinggi, atau penggunaan sistem interupsi yang cepat. Untuk sistem interupsi terdapat 3(tiga) buah mode, dan pada Z80 juga ditambahkan 2(dua) buah indeks register yaitu register 16-bit IX dan IY serta dilengkapi pula dengan sistem vektor interupsi melalui 8-bit IV register. Mengapa Z80 merupakan dasar prosesor yang populer adalah interface memori CPU dilengkapi dengan sinyal RAM refresh,

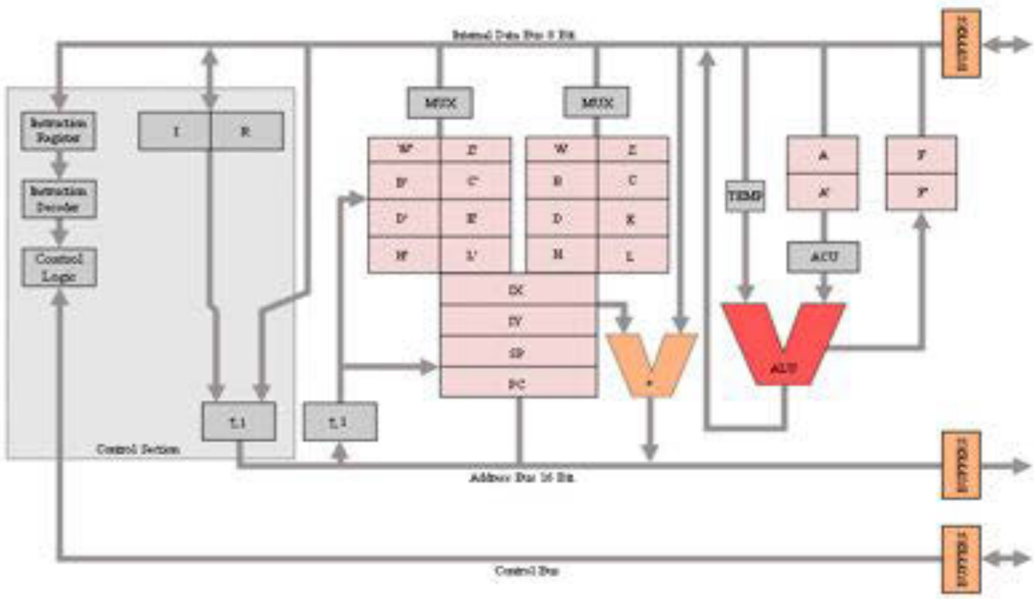
sehingga bagi pengembang banyak kemudahan dan harga yang murah. Z80 memiliki kompatibel yang tinggi terhadap 8080 dan CP/M yang merupakan standar sistem operasi mikroprosesor yang pertama dan terdapat beberapa produksi pilihan yaitu versi original Z80 (2.5 MHz), Zilog Z80A (4 MHz), Zilog Z80B (6MHz) dan Zilog Z80H (8 MHz). Z80 telah digunakan pada komputer game pada generasi pertama Nintendo Game Boy. Z80 produksi Sharp telah digunakan pada GameBoy Color, running pada clock 4 MHz untuk GameBoy software atau pada 8 MHz untuk Game Boy Color software. Z80 juga sukses digunakan dalam Sega Master System dan Game Gear. Z80 juga digunakan dalam Sega Genesis System untuk hardware reverse yang kompatibel dengan Sega Master System melalui cartridge khusus.



Gambar 1.7. IC Intel 8080 dan Goldstar Z8400A 4MHz/40-pin plastic DIP



Gambar 1.8. Arsitektur Intel 8080



Gambar 1.9. Arsitektur Zilog Z80

Tabel 1.1. Spesifikasi Prosesor

Z80	Famili 8080	Zilog terbaru
Z80 » 8-bit Mikroprosesor » Up to 20 MHz » 64 KB RAM » Single voltage » 256 I/O ports » 8080 emulation mode » 40-pin DIP	**8080** » Up to 3.1 MHz **8085** » Up to 8 MHz » Single voltage » On-chip peripherals	**Z800** » 8 / 16 bit CPU » Up to 16 MB RAM » 256 byte L1 cache » User / system modes » Z80 object-code compatible » Never released **Z8000** » 16-bit CPU » Up to 8 MB RAM » Normal / system modes » Multiprocessing » Virtual memory

4. Era Generasi Mikroprosessor

Era genesai Mikroprosesor berdasarkan tahun perkembangan dan pembuatan-nya dapat digambarkan sebagai berikut:

1971 : 4004 Mikroprosesor



Pada tahun 1971 perusahaan Intel untuk pertama kalinya memproduksi mikroprosesor yang diberi serial mikroprosesor 4004, aplikasi dari mikroprosesor tersebut digunakan pada sebuah kalkulator. Pada tahun inilah merupakan tonggak awal perkembangan mikroprosesor, yaitu mulai berkembangnya baik secara hardware maupun software.

1972 : 8008 Mikroprosesor

Pada tahun berikutnya yaitu tahun 1972 mikroprosesor dengan tipe mikro-prosesor 8008, dan mikroprosesor ini dikenal dengan keluarga 80xx yang memiliki teknologi arsitektur lebih baik dan kemampuan lebih dibanding mikroprosesor sebelumnya.

1974 : 8080 Mikroprosesor

Siemens ambil bagian dalam memproduksi Mikroprosesor tipe ini yaitu dengan serial SAB-nya, bahkan juga mengeluarkan produk trainer dengan nama Sitrain. Bersamaan mikroprosesor inilah diproduksi oleh Zilog Mikroprosesor dengan seri Z80, dan keduanya menjadi CPU dalam sistem komputer. Beberapa pabrik komputer diantaranya commodore, radioshack, apple, siclair dan bebrapa pabrik lainnya.

1978 : 8086-8088 Mikroprosesor

Pada masa ini IBM mulai memproduksi komputer menggunakan Mikroprosesor 8086-8088 dengan co-prosesor 8087 (tipe XT), sebuah era perkembangan luar biasa saat itu karena rancangan komputer adalah komputer personal. Dengan munculnya komputer pribadi buatan IBM penjualan komputer makin diminati baik kalangan perkantoran, bisnis dan pribadi. Saat itu IBM menjadi terkenal dan memiliki peluang bisnis yang luar biasa, saat itu pula IBM membntuk tim training komputer bersama penjualan produknya.

1982 : 80286 Microprocessor

Tipe AT dengan Mikroprosesor 80286 dengan co_prosesor 287 diproduksi Intel 286 dengan sistem operasi DOS, dilengkapi dengan sistem harddisk dan mulai dikembangkannya sistem operasi windows oleh Microsoft. Pada era inilah banyak bermunculan software komputer dan memberikan dominasi



sehingga perkembangan hardware harus mampu mengejar kebutuhan software.

1985 : Intel 80386™ Mikroprosesor

Sebuah prosesor Intel 80386 mulai digunakan dalam sistem komputer, prosesor ini dibangun dengan memanfaatkan sejumlah 275.000 transistor. Artinya dari sisi teknologi sudah lebih maju, karena penggunaan transistor dalam chip IC jauh lebih sedikit yaitu kurang lebih 100 kali lipat dibandingkan dengan Mikroprosesor 4004, pada generasi ini sudah berkembang sistem operasi windows yang dilengkapi sistem jaringan.

1989 : Intel 80486 DX CPU Mikroprosesor

Mikroprosesor ini memiliki kecepatan clock yang jauh lebih baik dibanding generasi sebelumnya, di sisi lain Windows memunculkan fasilitas sistem jaringan dengan servernya dan demikian juga Novel menjadi sistem jaringan yang digunakan hampir dipelosok dunia. Banyak kemudahan pemakaian berbagai aplikasi, muncul istilah "*what's you see, what's you get*" bahkan dengan adanya sistem mouse operasi aplikasi menjadi mudah dengan adanya fasilitas klik dan scroll. Dengan adanya co-prosesor yang semakib canggih fungsi matematika yang berane ragam rumus komplek dapat dilakukan, hal ini sangat meringankan dan memperkecil beban kerja pada processor.

1993 : Intel® Pentium® Processor

Mikroprosesor generasi yang dilengkapi dengan fasilitas multimedia sehingga mampu menangani berbagai jenis data seperti vois, bunyi, tulisan tangan, dan image.

1995 : Intel® Pentium® Pro Processor

Mikroprosesor sebagai perkembangan dari Intel® Pentium® Processor, fokus perkembangan adalah pada aplikasi server dan workstation. Pada era ini sistem jaringan sudah menjadi syarat perlu keberadaannya, yang diterapkan pada dunia bisnis untuk memproses data secara cepat, Mikroprosesor ini merupakan generasi yang luar biasa karena semua sektor mulai menggunakannya, mulai dari sistem bisnis sampai dunia pendidikan. .



1997 : Intel® Pentium® II Processo

Mikroprosesor Pentium II merupakan Mikroprosesor yang fokus pada pengembangan multimedia, yaitu dengan Intel MMX yang dirancang khusus untuk mengolah data video, audio, dan grafik. Sekitar 7 juta transistor digunakan untuk tipe Mikroprosesor ini, yang terintegrasi di dalamnya dengan Mikroprosesor disamping kemampuan tinggi dalam mengolah berbagai data dikembangkan pula untuk aplikasi internet.

1998 : Intel® Pentium II Xeon® Processor

Mikroprosesor yang dibuat untuk kebutuhan pada aplikasi server, segmen pasar yang dibidik adalah sektor bisnis yang lebih luas dengan kemampuan daya simpan data tinggi. Deangan berbasis pada Mikroprosesor inilah Intel dapat mencapai segmen pasar tertentu, sesuai dengan visinya yaitu memenuhi dan memberikan sebuah Mikroprosesor khusus.

1999 : Intel® Celeron® Processor

Mikroprosesor Intel Celeron merupakan Mikroprosesor yang memiliki kemampuan terbatas, ditujukan untuk kalangan pengguna yang tidak terlalu membutuhkan kinerja Mikroprosesor dengan kecepatan tinggi. Pemikiran sebuah komputer dengan harga murah dan dengan kemampuan cukup baik, merupakan strategi penjualan dalam bisnis komputer.

Perkembangan pengguna komputer untuk memenuhi kebutuhan hidupnya makin terasa, banyak dunia industri yang ingin membangun sebuah sistem komputer dengan kemampuan biaya yang tidak terlalu besar.

Dari segi rancangan arsitektur Mikroprosesor Intel Celeron ini memiliki bentuk yang mirip sama dengan Mikroprosesor Intel Pentium, kemampuan instruksi terbatas, L2 cache-nya lebih kecil, kecepatan clock yang lebih lambat, akan tetapi dengan harga terjangkau artinya harga yang jauh lebih murah dibanding Mikroprosesor Intel jenis Pentium. Dengan Mikroprosesor Celeron sektor pasar tertentu dapat dimasuki, disini bisnis komputer benar-benar menjadi pertimbangan sebagai sektor yang jumlah pengguna lebih banyak.

1999 : Mikroprosesor Intel® Pentium® III

Mikroprosesor Pentium III merupakan Mikroprosesor yang memiliki jumlah 70 instruksi baru lebih banyak, yang dikembangkan untuk mendukung dan memperkaya kemampuan audio streaming, pengolahan citra, aplikasi tiga dimensi, dan mendukung perkembangan dunia video serta pengenalan suara.

1999 : Mikroprosesor Intel® Pentium® III Xeon®

Tren baru dengan membuat peningkatan kemampuan produk, Intel kembali berupaya konsentrasi pada pangsa pasar untuk server dan workstation. Untuk itu Intel memasarkan seri Xeon jenis Pentium III dengan kemampuan instruksi SIMD sejumalah 70 perintah. Mikroprosesor ini memiliki kemampuan kecepatan transfortasi informasi yang tinggi melalui sistem bus ke processor, artinya kecepatan proses pada sistem komputer menjadi meningkat secara signifikan.

Untuk saling berkomunikasi dengan Mikroprosesor lain, maka Mikroprosesor ini juga difasilitasi untuk dapat saling bekerja sama dengan Mikroprosesor lain sejenis.

2000 : Mikroprosesor Intel® Pentium® IV

Mikroprosesor Pentium IV merupakan produk Intel, dimana kecepatan prosesnya mampu bekerja pada kecepatan hingga 3.06 GHz. Runtutan produk pertama kali keluar Mikroprosesor berkecepatan 1.5 GHz dengan konfigurasi pin 423, kemudian dikembangkan dengan konfigurasi 478. Mikroprosesor ini dikembangkan mulai dengan processor Intel Pentium IV berkecepatan 1.3 GHz, dan dikembangkan terus dengan kecepatan hingga 3.4 GHz.

2001 : Mikroprosesor Intel® Itanium®

Mikroprosesor jenis Itanium merupakan Mikroprosesor pertama berbasis 64 bit, yang ditujukan untuk memenuhi kebutuhan pasar akan server dan workstation atau pemakai khusu. Terdapat perubahan besar pada Mikroprosesor ini, karena dibangun dengan arsitektur baru yang benar-benar

berbeda dari generasi sebelumnya. Teknologi yang digunakan didasarkan pada rancangan teknologi EPIC yaitu *Intel's Explicitly Parallel Instruction Computing*. Selanjutnya Mikroprosesor ini dikembangkan dengan seri Itanium II adalah generasi kedua dari keluarga Itanium pada tahun berikutnya.

2003 : Mikroprosesor Intel® Pentium® M

Berbasis pada komputer mobil, artinya ringan dan mudah dibawa kemana saja sesuai kebutuhan pengguna, Intel® PRO/WIRELESS 2100 dengan komponen Intel® Centrino™ dipakai sebagai teknologi untuk memenuhi pasar ini. Berarti Intel Centrino dibuat sebagai sebuah komputer yang mudah dibawa kemana-mana untuk memenuhi kebutuhan pasar yang semakin luas.

2004 : Mikroprosesor Intel E7520/E7320 Chipsets

Mikroprosesor ini merupakan dual processor dengan konfigurasi 800MHz FSB, DDR2 400 memory, and PCI Express peripheral interfaces.

2005 : Mikroprosesor Intel Pentium 4 Extreme Edition 3.73GHz

Sebuah Mikroprosesor yang ditujukan untuk pasar tertentu, Mikroprosesor ini menggunakan konfigurasi 3.73GHz frequency, 1.066GHz FSB, EM64T, 2MB L2 cache.

2005 : Mikroprosesor Intel Pentium D 820/830/840

Mikroprosesor 64 bit lahir dengan sebutan dual core, yang di dalamnya menggunakan 2 buah core, konfigurasi 1MB L2 cache pada tiap core, 800MHz FSB, dengan clock frekuensi 2.8GHz, 3.0GHz, dan 3.2GHz, HyperThreading.

2006 : Mikroprosesor Intel Core 2 Quad Q6600

Mikroprosesor tipe desktop memiliki 2 buah core dengan konfigurasi 2.4GHz dengan 8MB L2 cache, 1.06GHz thermal design power (TDP).

2006 : Mikroprosesor Intel Quad-core Xeon X3210/X3220

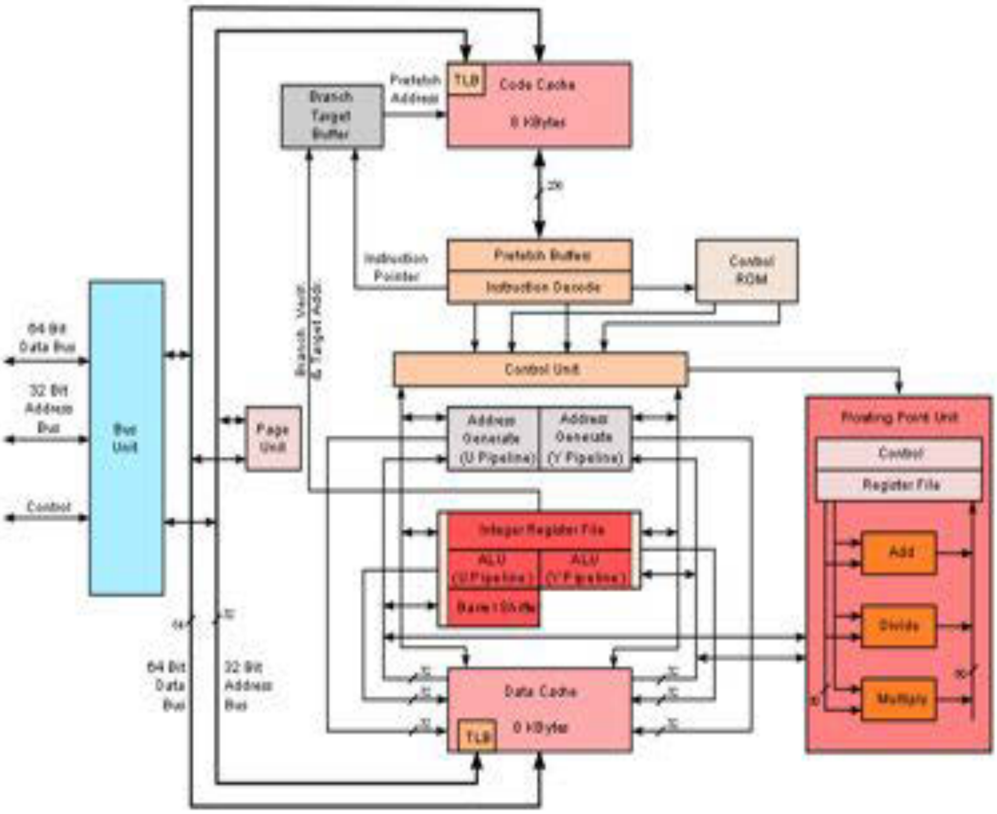
Mikroprosesor tipe server dan memiliki 2 buah core dengan konfigurasi sistem clock 2.13 dan 2.4GHz, 8MB L2 cache 1.06GHz Front-side bus, dan thermal design power (TDP)

5. Era Generasi Intel Pentium

Hingga saat ini Intelpun melakukan pengembangan dan ada banyak macam prosesor yang tersedia di pasaran saat ini, rancangan atau desain terus berkembangan untuk memenuhi kebutuhan kehidupan. Untuk saat ini sangat dibutuhkan komputer yang memiliki fasilitas multimedia dan komputer menjadi komputer portable, dan Intel selalu berusaha meningkatkan fasilitas dari prosesor yang digunakan. Berikut tahap pengembangan yang dilakukan Intel yang mampu mengendalikan pasar:

6. Teknologi MMX

Intel mengembangkan teknologi MMX yang dilengkapi dengan 57 instruksi baru, kecepatan akses yang lebih tinggi dari generasi sebelumnya, peningkatan fasilitas sehingga dapat digunakan untuk memenuhi fitur-fitur pemrograman multimedia dan komunikasi. Hal tersebut pada generasi sebelumnya dilakukan dengan beberapa prosesor dalam mengimplementasikan pemrograman komunikasi data dan aplikasi suara. Secara hardware ada peningkatan dari sisi jumlah register pendukung prosesor dalam melakukan operasinya, yaitu sejumlah 64-bit register di tambahkan ke dalam chip prosesor.



Gambar 1.10. Arsitektur Intel P5

Core Prosesor Pentium pertama oleh intel diberi kode P5, dan produk ini menggunakan mikroprosesor tipe 80501 yang sebelumnya digunakan tipe 80500. Terdapat dua versi dengan spesifikasi operasi clock 60 MHz dan 66 MHz, tegangan yang digunakan 5 volt seperti layaknya tegangan yang digunakan pada chip IC TTL. Dihitung dari transistor yang terpasang dalam chip IC-nya mencapai 3,1 juta dan dikemas dalam ukuran 16,7x17,6 mm yang berarti luas areanya adalah 293.92 mm^2 dan dibangun dalam proses 0.8 µm BiCMOS.

Arsitektur haradware P5 adalah berbasis pada Intel 8080 dengan penambahan kecepatan clock dan kemampuan untuk memfasilitasi komunikasi data, multi media oleh karena itu dikenal dengan teknologi MMX (Multi Media Extenstion). Berikut merupakan gambar arsitektur P5.

Peningkatan kecepatan operasi dibuat oleh intel dengan produknya P54C (80502, prosesor ini bekerja engan operasi clock 75, 90, atau 100 MHz menggunakan sumber tegangan 3.3 volt. Prosesor ini merupakan generasi pertama yang menggunakan tegangan 3,3 volt sehingga dapat mereduksi pemakaian daya listrik. Untuk memberikan jaminan rangkaian internal bekerja dengan frekuensi lebih tinggi dari P5 maka prosesor ini dilengkapi dengan sebuah nternal clock multiplier, dengan demikian akan berdampak pula untuk operasi external address dan bus data yang makin komplek. Hal inipun memungkinkan penerapan teknologi two-way multiprocessing, sistem interupsi memiliki fitur baru pengelolaan interupsi sebagaimana diberlakukan pada chip 8259. Jumlah transistor di dalamnya mencapai 3,3 juta dan dalam ukuran kemasan 163 mm^2. Pabrikasi dilakukan dengan teknologi proses BiCMOS yang dilakukan dengan ketebalan 0.5 µm dan 0.6 µm.

Pengembangan P54C diikuti tipe berikutnya yaitu P54CQS yang beroperasi pada 120 MHz, dan tipe berikutnya P54CQS dikuti tipe P54CS yang beroperasi pada 133, 150, 166 dan 200 MHz. Tipe MMX berikutnya adalah P55C (80503) yang dikembangkan oleh Intel's Research & Development Center di Haifa, Israel, dan menjualnya di pasaran dengan sebutan Pentium with MMX Technology yang selanjutnya lebih dikenal dengan Pentium MMX, walaupun dasar pengembangan adalah P5 namun demikian fitur ditambahkan instruksi baru sejumlah 57 instruksi MMX dengan maksud meningkatkan performansi tugas-tugas multimedia, seperti encoding dan decoding digital media data (Oktober 1996).

Instruksi pada prosesor ini bekerja dengan tipe data paket 64-bit, meliputi integer 8-bit, 4 integer 16-bit, dua integer 32-bit, atau satu integer 64-bit. Sebagai contoh instruksi asembler PADDUSB (Packed ADD Unsigned Saturated Byte) merupakan instruksi penjumlahan dua vektor, dimana setiap vektor berisi 8-bit unsigned integer. Berisi 4,5 juta trasistor dan dengan ukuran luas 140 mm^2, pabrikasi dilakukan dengan teknologi proses 0.28 µm CMOS.

Untuk motherboard yang digunakan P55C adalah kompatibel dengan konfigurasi motherboard Soket 7, dan Soket 7 spesifikasi secara fisik dan



elektris diperuntukan CPU x86-style pada komputer personal motherboard serta dipersiapkan untuk CPU dari berbagai merk.

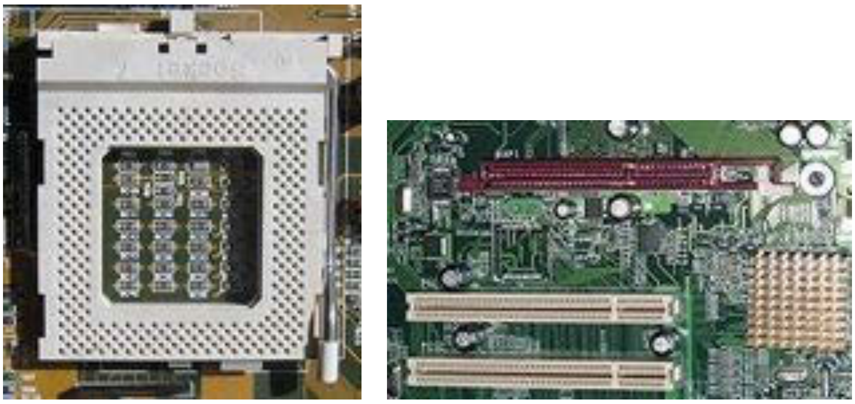
Perbedaan antara soket 5 dan soket 7 adalah pada soket 7 dilengkapi dengan layanan dual split rail voltage sedangkan pada soket 5 diberlakukan single voltage, namun demikian tidak semua perusahaan motherboard menyediakan fasilitas dual voltage. Untuk soket 7 kompatibel dengan CPU soket 5, sehingga soket 5 CPU dapat diletakan pada motherboard soket 7.

Prosesor yang menggunakan standar soket 7 diantaranya adalah AMD K5 dan K6, Cyrix 6x86 dan 6x86MX, IDT WinChip, Intel P5 Pentium (2.5–3.5 V, 75–200 MHz), Pentium MMX (166–233 MHz), dan Rise Technology mP6. Untuk AMD Geode LX dan Geode GX tetap menggunakan soket 7. Soket 7 memiliki 321-pin (19 x 19 pin) dikenal dengan soket SPGA ZIF atau soket SPGA LIF dengan 296-pin (37x37 pin) tetapi jarang digunakan.

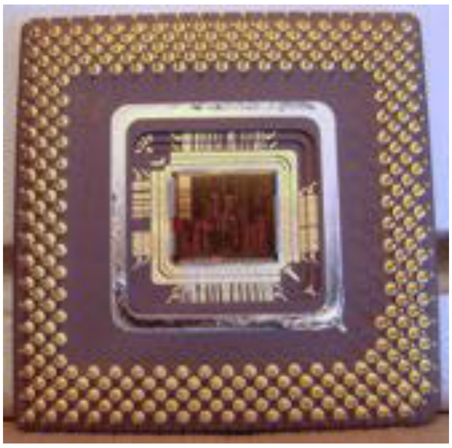
Ekstensi dari sket 7 adalah super soket 7 yang dikembangkan AMD untuk produksinya prosesor K6-2 dan K6-III, untuk operasi clock yang lebih tinggi digunakan AGP. Accelerated Graphics Port (sering disingkat dengan AGP) merupakan high-speed point-to-point channel yang berfungsi untuk menempatkan video card ke motherboard komputer, terutama digunakan untuk membantu percepatan 3D computer graphics.



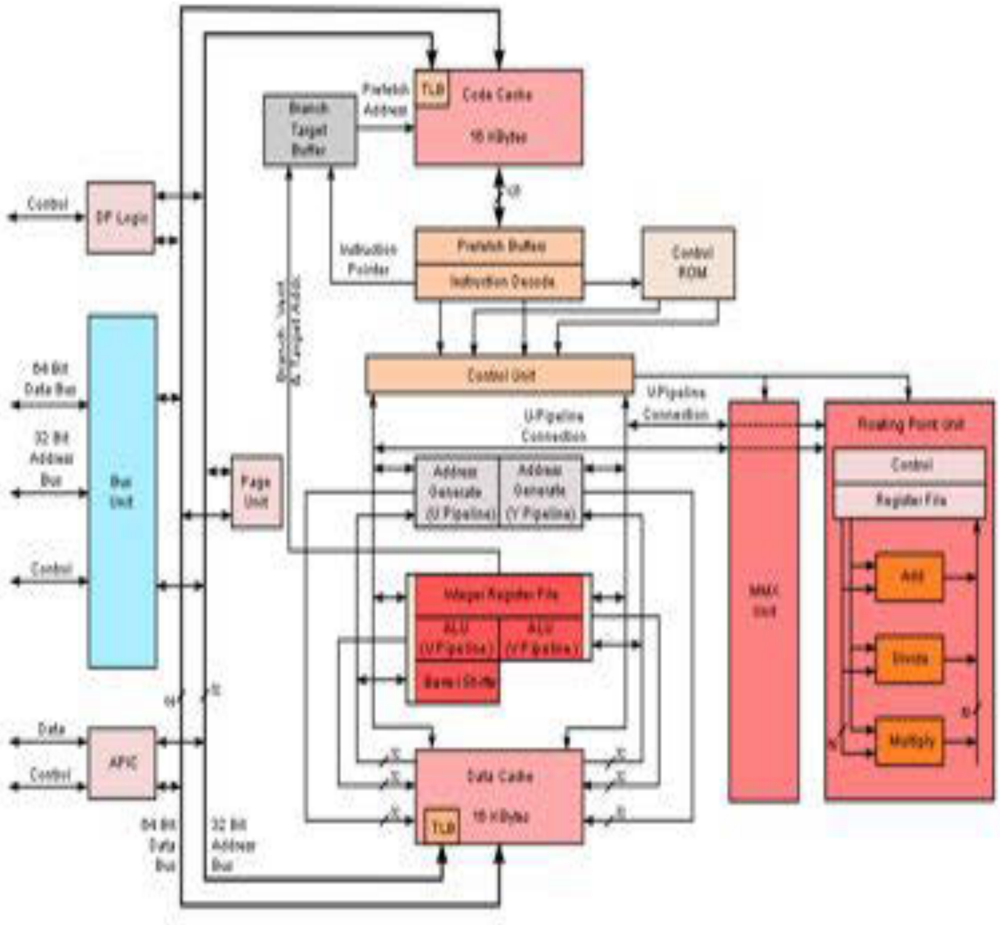




Gambar 1.11. Soket 7



Gambar 1.12. Pentium MMX-166 MHz.



Gambar 1.13, Arsitektur Intel MMX

7. Pentium II

Peningkatan kecepatan clock dari prosesor menjadi target berikutnya, yaitu dengan dikeluarkannya Prosessor yang memiliki fitur kecepatan antara 233MHz sampai 450MHz (di tahun 1999). Hal tersebut mendorong pengguna untuk pemanfaatan lebih luas karena prosesor ini dapat dikonfigurasi untk dapat berfungsi sebagai workstations maupun servers, ditunjang perkembangan sistem operasi Windows Workgroup. Dari sisi hardware diterapkan teknologi single edge contact cartridge (242 pin) 512KB, level two cache 32KB dengan pembagian 16KB data dan 16KB instruksi cache



Gambar 1.14. Intel 486 DX2

8. Pentium Pro III

Kebutuhan akan sistem jaringan dengan kemampuan server yang tinggi mendorong munculnya prosesor tipe ini, sehingga rancangan atau desain prosesor diarahkan untuk dapat mencapai high-end server yang operasinya dibutuhkan 4 buah Prosessor. Fitur dikembangkan untuk bisa digunakan sebagai high end workstation dan server dengan kecepatan sampai 200MHz, prosesor ini memiliki kemampuan 4 (empat) kali kemampuan prosesor sebelumnya dan merupakan era prosesor 32 bit.

9. Prosesor Celeron

Prosessor Cerelon didesain untuk pemakaian pasar konsumen dirumahan. Prosessor ini memiliki fitur :

Kecepatan berkisar dari 266 sampai 500MHz (di tahun 1999) Mirip dengan Pentium II Prosessor Versi 300 dan 333MHz termasuk 128K dari level two cache level one cache 32K (terdiri dari 16K instruksi dan 16K data) meliputi teknologi MMX

10. Pentium III Prosessor

Berdasarkan pada mikro arsitektur P6, merupakan media Intel MMX yangditingkatkan dengan penyediaan Streaming SIMD Extensions. Diamanterdapat 70 instruksi baru yang memungkinkan penggambaran imagetingkat lanjut, grafik 3D, audio dan video, dan pengenalan percakapan.Fitur barunya adalah Prosessor serial number, yaitu suatu

nomerelektronik yang ditambahkan ke setiap Prosessor Pentium III, yang dapat digunakan oleh departement IT untuk manajemen informasi/ asset.Prosessor ini memiliki fitur kecepatan berkisar450MHz, 500MHz, 550MHz dan 600MHz (di tahun1999), 70 Instruksi baru, P6 Microarchitecture, 100MHz system bus, 512K Level Two Cache, Intel® 440BX chipset

11. Xeon Pentium III Prosessor

Merupakan Prosessor yang dapat diskalakan (multiProsessor) sebanyak 2, 4, 8 atau lebih dan didesain secara khusus untuk mid-range dan server workstations yang lebih tinggi tingkatannya. Prosessor ini memiliki fitur sesuai untuk high end workstations atau high end servers, kecepatan berkisar dari 500 sampai 550MHz (di tahun 1999), mendukung penskalaan multiProsessor, Memiliki Prosessor serial number, 32KB (16KB data /16KB instruction) nonblocking, L1 cache, 512 Kbytes L2 cache B.

1.3 Perbedaan Antar Mikroprosessor

1. Komparasi 8085 Dengan Z80

No.	8085 Mikroprosesor	Z80 Mikroprosesor
1	Jalur data multipleksi	Jalur data tidak ada multipleksi
2	74 instruksi	158 instruksi
3	Operasi pada 3 - 5 MHz	Operasi pada 4 - 20 MHz
4	Memiliki 5 Interrupsi	Memiliki 2 Interrupsi
5	Tidak memiliki dynamic memory pada board	Memiliki dynamic memory pada board untuk refresh Dynamic memory
6	Tidak dilengkapi register Index.	dilengkapi dengan 2 register Index.
7	Berisi SIM & RIM	Tidak memiliki SIM & RIM



2. Komparasi 8085 Dengan 6800

No.	8085 Mikroprosesor	MC6800 Mikroprosesor
1	Operasi pada frekuensi clock 3 - 5 MHz.	Operasi dengan frekuensi clock 1 MHz
2	8085 tidak dilengkapi register Index.	dilengkapi register Index.
3	8085 tidak dilengkapi rangkaian logik clock pada board.	tidak dilengkapi rangkaian logik clock pada board.
4	8085 memiliki satu akumulator.	MC6800 memiliki dua akumulator
5	8085 memiliki 5 interrupsi.	MC 6800 memiliki 2 interrupsi
6	Jumlah instruksi total 674.	MC6800 Jumlah instruksi total 72.

3. Komparasi 8085 Dengan 80386

No.	8085 Mikroprosesor	80386 Mikroprosesor
1	Merupakan 16 bit Mikroprosesor dan generasi pertama 16 bit Mikroprosesor setelah 8085(8-bit).	Merupakan 32 bit Mikroprosesor dan merupakan ekstensi logika dari 80236.
2	Memiliki arsitektur jalur pipa (*pipeline*) bukan level tinggi dan memiliki bus interface kecepatan tinggi (high speed) dalam satu chip.	Memiliki arsitektur jalur pipa level tinggi dan bus jauh lebih dibanding 8086.



3	Merupakan panutan utama kompatibel dengan 80386, artinya semua instruksi 8086 diikuti oleh 80386.	Selain 80386 dapat menunjang model pemrograman pada 8086, dapat juga program yang ditulis untuk 8086 dalam mode virtual yaitu jika VM=1 (protected mode) langsung dijalankan.
4	Dikemas dalam paket DIP 40 pin	Chip IC dari 80836 berisi 132 pin.
5	Dibangun dari teknologi HMOS.	Dibangun menggunakan teknologi *High-speed CHMOS III.*
6	Tidak dibutuhkan hardware khusus untuk tugas-tugas Switching.	Dibutuhkan hardware khusus untuk tugas-tugas Switching.
7	8086 beroperasi dengan Clock 5MHz..	80386 beroperasi dengan frekuensi maksimum Clock 33MHz.
8	Bus address dan bus data bus multiplek.	Jalur terpisah antara address dan data bus untuk hemat waktu.
9	Memiliki paket transistor sejumlah 29.500 buah transistor.	Jumlah transistor dan kompleksitas meningkat sampai 2.75.000 buah.
10	Jumlah instruksi yang dimiliki 117 buah.	Jumlah instruksi yang dimiliki 129 buah.
11	Tidak dilengkapi dengan proteksi mekanik (*paging*).	Pada 80386 dilengkapi dengan proteksi mekanik (*paging*), dengan instruksi untuk mendukungnya.



12	Beroperasi hanya dalam satu mode.	Beroperasi dengan 3 (tiga) mode: a) Real b)Virtual c)Proteksi
13	Hanya memiliki antrian instruksi.	Memiliki antrian instruksi seperti halnya pada *pre fetch queue*.
14	Dalam 8086, tidak semua operasi dalam mode parallel.	80386 semua unit fungsional tidak ada yang parallel
15	8086 memiliki 9 (sembilan) flag.	Memiliki semua kesembilan flag 8086 tetapi flag lainnya diberi nama IOP,NT,RF,VM.

4. Komparasi 8086 Dengan 8088

No.	8086 Mikroprosesor	8088 Mikroprosesor
1	Antrian instruksi adalah 6 byte.	Antrian instruksi adalah 4 byte.
2	Memori dalam 8086 dibagi menjadi 2(dua) bank, yang terdiri dari 1,048,576 byte.	Memori dalam 8088 tidak dibagi menjadi 2(dua) bank seperti halnya pada 8086.
3	Data bus pada 8086 adalah 16-bit	Data bus pada 8088 adalah 8-bit
4	Memiliki sinyal BHE(bar) pada pin no. 34 dan tidak ada sinyal SSO(bar).	Tidak memiliki sinyal BHE(bar) pada pin no. 34 dan hanya ada sinyal SSO (bar) serta tidak ada pin S7.



5	Sinyal output digunakan untuk memilih memori atau I/O pada M/IO(bar) , tetapi jika IO(bar)/M pada kondisi low atau logik '0' berarti memilih devais I/O dan jika IO(bar)/M pada kondisi high atau berlogika '1' berarti memilih memori.	Sinyal output digunakan untuk memilih memori atau I/O pada M/IO(bar) , tetapi jika IO(bar)/M pada kondisi low atau logik '0' berarti memilih memori dan jika IO(bar)/M pada kondisi high atau berlogika '1' berarti memilih devais I/O.
6	Membutuhkan satu siklus mesin untuk sinyal R/W jika pada lokasi even dan untuk yang lain dibutuhkan dua.	Membutuhkan satu siklus mesin untuk sinyal R/W jika pada lokasi even dan untuk yang lain dibutuhkan dua.
7	Dalam 8086, semua bus address & data adalah multiplek.	Dalam 8088, bus address, AD_7-AD_0 adalah multiplek.
8	Dibutuhkan dua IC 74343 de-multiplek AD_0-AD_{19}.	Dibutuhkan satu IC 74343 de-multiplek AD_0-AD_7.

1.4. Latihan

Tugas 1.

1. Bacalah dan pahami dengan seksama uraian-uraian materi pada kegiatan belajar **Memahami Teknologi Pertama Mikroprosessor**. Bila ada materi yang kurang jelas, siswa dapat bertanya pada guru atau instruktur pengampu kegiatan belajar.
2. Buat catatan-catatan kecil sebagai kata kunci untuk setiap materi pokok.
3. Bermodalkan poin 1 dan poin 2, lakukan diskusi kelompok yang terdiri maksimal 4 orang peserta dalam satu kelompok.
4. Buatlah laporan hasil diskusi dalam bentuk ringkasan pemahaman bersama pada kelompok anda.



5. Untuk materi yang belum jelas, diskusikan dengan instruktur atau pengampu mata pelajaran.

Tugas 2.

1. Bacalah dan pahami dengan seksama uraian-uraian materi pada kegiatan belajar **Memahami Teknologi IC pada Mikroprosessor** . Bila ada materi yang kurang jelas, siswa dapat bertanya pada guru atau instruktur pengampu kegiatan belajar.
2. Buat catatan-catatan kecil sebagai kata kunci untuk setiap materi pokok.
3. Bermodalkan poin 1 dan poin 2, lakukan diskusi kelompok yang terdiri maksimal 4 orang peserta dalam satu kelompok.
4. Buatlah laporan hasil diskusi dalam bentuk ringkasan pemahaman bersama pada kelompok anda.
5. Untuk materi yang belum jelas, diskusikan dengan instruktur atau pengampu mata pelajaran.

Tugas 3.

1. Bacalah dan pahami dengan seksama uraian-uraian materi pada kegiatan belajar **Memahami Perkembangan Arsitektur Mikroprosessor.** Bila ada materi yang kurang jelas, siswa dapat bertanya pada guru atau instruktur pengampu kegiatan belajar.
2. Buat catatan-catatan kecil sebagai kata kunci untuk setiap materi pokok.
3. Bermodalkan poin 1 dan poin 2, lakukan diskusi kelompok yang terdiri maksimal 4 orang peserta dalam satu kelompok.
4. Buatlah laporan hasil diskusi dalam bentuk ringkasan pemahaman bersama pada kelompok anda.

Untuk materi yang belum jelas, diskusikan dengan instruktur atau pengampu mata pelajaran

Tugas 4.

1. Bacalah dan pahami dengan seksama uraian-uraian materi pada kegiatan belajar **Memahami Era Generasi Mikroprosessor**. . Bila ada materi yang kurang jelas, siswa dapat bertanya pada guru atau instruktur pengampu kegiatan belajar.
2. Buat catatan-catatan kecil sebagai kata kunci untuk setiap materi pokok.
3. Bermodalkan poin 1 dan poin 2, lakukan diskusi kelompok yang terdiri maksimal 4 orang peserta dalam satu kelompok.
4. Buatlah laporan hasil diskusi dalam bentuk ringkasan pemahaman bersama pada kelompok anda.
5. Untuk materi yang belum jelas, diskusikan dengan instruktur atau pengampu mata pelajaran.

Tugas 5.

1. Bacalah dan pahami dengan seksama uraian-uraian materi pada kegiatan belajar **Memahami Perbedaan Antar Mikroprosessor**. Bila ada materi yang kurang jelas, siswa dapat bertanya pada guru atau instruktur pengampu kegiatan belajar.
2. Buat catatan-catatan kecil sebagai kata kunci untuk setiap materi pokok.
3. Bermodalkan poin 1 dan poin 2, lakukan diskusi kelompok yang terdiri maksimal 4 orang peserta dalam satu kelompok.
4. Buatlah laporan hasil diskusi dalam bentuk ringkasan pemahaman bersama pada kelompok anda.



5. Untuk materi yang belum jelas, diskusikan dengan instruktur atau pengampu mata pelajaran.



SISTEM MIKROPROSESOR

DESKRIPSI MATERI PEMBELAJARAN

Penerapan mikroprosesor sudah melekat pada kehidupan manusia mulai dari peralatan rumah tangga, sistem keamanan mobil dan rumah terlebih lagi untuk sistem kendali di bidang militer dan bidang industri. Sistem kerja mikroprosesor dalam penerapan sebagai pusat pengolah data dan pengendali didukung oleh macam-macam komponen di dalamnya, oleh karena itu untuk dapat menerapkan mikroprosesor perlu memahami komponen beserta fungsinya. Sedangkan sistem mikroprosesor meliputi bus, *memory map* dan *address decoder*, memori, *pheriperal input-output*, dan dasar pembentukan komponen adalah dibangun dari rangkaian gerbang-gerbang digital. Untuk dapat menerapkan sistem mikroprosesor perlu desain rangkaian minimal sistem, sehingga mikroprosesor tersebut dapat berfungsi sebagai pusat pengolah data dan pengendali.



KOMPETENSI INTI (KI-3)

Kompetensi Dasar (KD):

2. Menerapkan macam-macam komponen sistem mikroprosesor

Indikator:

1.3. Memahami macam-macam komponen sistem mikroprosesor

1.4. Merencanakan sistem mikroprosesor meliputi bus, *memory map* dan *address decoder*, memori, *pheriperal input-output*

1.5. Mendesain sirkuit diubah menjadi tata letak komponen

KOMPETENSI INTI (KI-4)

Kompetensi Dasar (KD):

2. Melakukan eksperimen sistem mikropro-sesor

Indikator:

1.3. Melakukan eksperimen sistem mikro-prosesor dan interprestasi data hasil pengukuran.

1.4. Melakukan eksperimen sistem mikro-prosesor meliputi bus, memory map dan adress decoder, memori, pheri-peral input-output serta interprestasi data hasil pengukuran

1.5. Membuat diagram rangkaian (sirkuit) menjadi tata letak komponen.

KATA KUNCI PENTING

- mikroprosesor
- bus, *memory* dan *address decoder*
- *pheriperal input-output*

BAB II. KOMPONEN SISTEM MIKROPROSESOR

2.1. DASAR KOMPONEN MIKROPROSESOR

Alam semesta, semua mahluk baik binatang ataupun tumbuhan dan manusia merupakan ciptaan Tuhan Yang Maha Kuasa, dari sekian banyak ciptaanNYA coba kita simak tubuh manusia. Dalam tubuh manusia terdiri dari berbagai macam organ misal jantung, hati, kepala, kaki, tangan dan organ tubuh lainnya yang kesemuanya membentuk sebuah sistem tubuh, secara kasat mata terlihat bahwa antar organ dalam membentuk sistem tubuh saling tergantung dan saling berhubungan sistem kerjanya. Setiap gerak organ selalu diawali adanya perintah dari otak berupa sinyal listrik yang dikirimkan melalui jalur kendali menuju organ yang diperintahkan untuk melaksanakan gerak, artinya setiap gerak organ melalui sinyal peritah dan gerak organ satu



dengan lainnya juga selalu ada sinkronisasi. Sedangkan organ dibentuk dari sel, dan setiap sel pembentuk organ memiliki kesamaan karakteristik dan prinsip kerja. Disisi lain sumber tenaga penggerak yang tersedia dalam tubuh manusia mampu memberikan tenaga selam 24 jam terus menerus. Dari hasil menyimak sedikit tentang tubuh manusia dilihat dari perspektif pembentuk sistem tubuh, maka sudah seharusnya kita bersyukur dan mau mengakui keagunganNYA.

Sistem mikroprosesor dikembangkan manusia juga mengikuti pola tubuh manusia, yaitu sistem dibuat dengan membentuk organ-organ yang sering istilahnya disebut dengan komponen. Setiap komponen pembentuk sistem mikroprosesor dibangun dari rangkaian-rangkaian gerbang digital, antar komponen dihubungkan dengan sistem jalur data (*Data BUS*), untuk menunjuk komponen digunakan jalur alamat (*Address Bus*) dan untuk sinkronisasi kerja antar komponen digunakan jalur kontrol (*Control Bus*). Seperti pada tubuh manusia pada sistem mikroprosesor untuk memanfaatkan atau menggunakan fungsi komponen dilakukan dengan perintah yang dikirim dari pusat pengolah yang disebut *Arithmatic Logic Unit* (ALU).

Untuk memahami komponen sistem mikroprosesor, berikut dijelaskan rangkaian-rangkaian gerbang digital pembentuk komponen tersebut.

2.1.1. Catu Daya Mikroprosesor

Sebuah mikroprosesor akan dapat bekerja dengan sempurna apabila diberikan tegangan listrik dari catu daya yang stabil pada tegangan kerjanya, oleh karena itu pada bab ini dijelaskan rangkaian dasar catu daya sesuai dengan tegangan yang dibutuhkan sistem mikroprosesor.

Catu daya yang dibutuhkan oleh sistem mikroprosesor umumnya satu sumber dengan tegangan listrik sebesar +5 volt yang diperuntukan khusus memberikan catu pada mikroprosesor, rangkaian digital pendukung berupa tegangan untuk komponen-komponen TTL, dan untuk rangkaian tertentu dengan tegangan +/- 12 volt seperti konversi digital ke analog serta untuk memori (RAM) dinamik dengan tegangan – 5 volt.

Sedangkan untuk kebutuhan tegangan yang diperuntukan rangkaian pemrogram EPROM dibutuhkan tegangan listrik antara 22 volt sampai 26 volt, kebutuhan arus listrik untuk keseluruhan sistem berkisar antara 2 ampere sampai 5 ampere hal ini tergantung dari kompleksitas rangkaian yang dibuat dalam sistem mikroprosesor.

a. Catu Daya 5 volt.

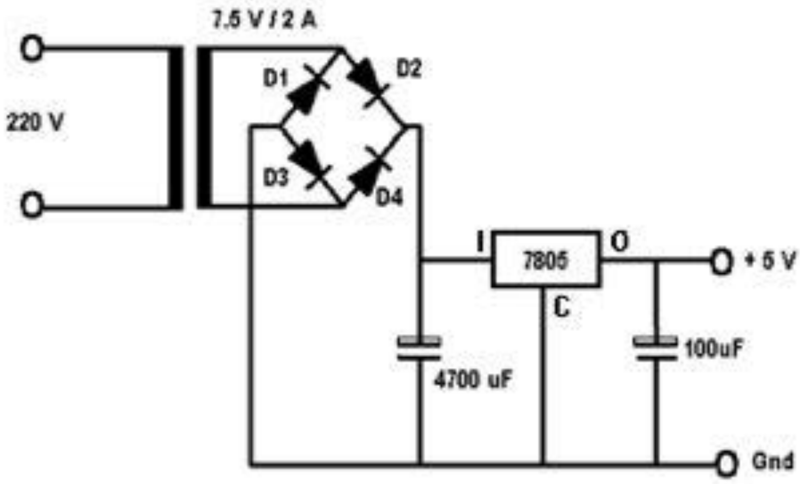
Catu daya 5 volt dapat dibangun dari rangkaian dasar penyearah 4 dioda yang membentuk model penyearah gelombang penuh, oleh karena dibutuhkan tegangan 5 volt dengan stabilitas tegangan yang tinggi maka dibutuhkan sebuah rangkaian penstabil tegangan.

Pada prinsipnya penstabil tegangan harus diberikan tegangan masukan lebih tinggi dari 5 volt, agar IC penstabil tegangan tidak terlalu panas maka tegangan masukan yang berasal dari sebuah transformator diambil dengan tegangan antara 7,5 volt sampai 9 volt serta kemampuan memberikan arus listrik sebesar 2 ampere.

Gambar 1 menunjukan sebuah rangkaian catu daya 5 volt dengan menggunakan penstabil tegangan 7805, dimana transformator berfungsi sebagai penurun tegangan dari 220 volt (AC) menjadi tegangan 7,5 volt (AC). Melalui 4 buah dioda yang terpasang secara metode jembatan tegangan tersebut disearahkan menjadi arus DC gelombang penuh, dengan memasang sebuah filter berupa kondensator 4700 uF/16 volt maka gelombang penuh tersebut dibuat menjadi arus listrik searah. Selanjutnya tegangan DC yang keluar dari filter (kondensator) dibuat agar stabil selalu mengeluarkan tegangan sebesar 5 volt (konstan), untuk itu dipasang sebuah IC penstabil tegangan dengan tipe 7805. Dengan memasang IC 7805 inilah didapatkan tegangan yang stabil sebesar 5 volt dan dapat menyediakan arus searah sebesar 2 ampere.

Yang perlu diperhatikan dalam rangkaian ini adalah jangan sampai rangkaian diode jembatan terbalik, pemasangan polaritas kondensator filter terbalik hal

ini akan menyebabkan kondensator meletus dan jangan sampai memasang IC penstabil tegangan tertukar kaki-kakinya.



Gambar 2.1. Catu Daya +5 volt / 2 ampere dengan penstabil tegangan IC 7805.

Rangkaian yang ditunjukan pada gambar 2.1 cukup baik untuk digunakan sebagai pemberi tegangan dan arus listrik pada sistem mikroprosesor, karena berdasarkan hasil pengujian dengan memberikan beban yang bervariasi mulai 0 ampere sampai 2 ampere ternyata tegangan tetap stabil + 5 volt.

Agar kualitas lebih baik lagi maka IC 7805 perlu diberi pendingin dari bahan aluminium, sehingga panas yang tinggi pada IC 7805 tidak terjadi lagi. Oleh karena sistem kerja clock mikroprosesor memiliki frekuensi tinggi maka perlu dipasangkan filter untuk frekuensi tinggi, untuk itu dua kondensator yang masing-masing besarnya 100 nF dipasangkan secara paralel pada kedua kondensator sebelumnya.

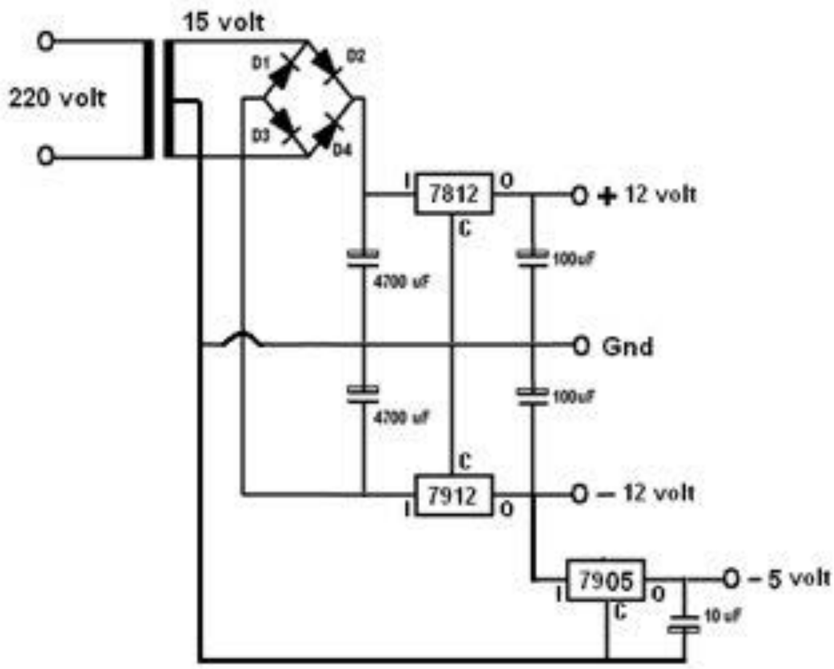
b. Catu Daya +/- 12 volt.

Catu daya +/- 12 volt dapat dibangun dari rangkaian dasar penyearah 4 dioda seperti pada catu daya 5 volt, keempat diode membentuk model penyearah gelombang penuh. Penyearah tersebut diumpankan pada kumparan sekunder transformator dengan tegangan bolak-balik 15 volt, oleh karena dibutuhkan

tegangan +/- 12 volt maka diperlukan titik tengah sebagai ground dan untuk menjaga stabilitas tegangan yang tinggi maka dipasangkan dua buah rangkaian penstabil tegangan positip dan penstabil tegangan negatip.

Dengan dipasangnya penstabil tegangan maka diperlukan tegangan masukan lebih tinggi dari 12 volt, untuk menjaga agar IC penstabil tegangan tidak terlalu panas maka tegangan masukan yang berasal dari transformator diambil dengan tegangan sebesar 15 volt untuk keperluan ini dibutuhkan arus listrik sebesar 1 sampai 2 ampere.

Gambar 2.2 menunjukan sebuah rangkaian catu daya +/- 12 volt dengan menggunakan penstabil tegangan 7812 untuk sumber tegangan positip dan 7912 untuk sumber tegangan negatip, dan transformator berfungsi sebagai penurun tegangan dari 220 volt (AC) menjadi tegangan 15 volt (AC) dengan titik tengah. Melalui 4 buah dioda yang terpasang secara metode jembatan tegangan tersebut disearahkan menjadi arus DC gelombang penuh, dengan memasang sebuah filter berupa kondensator 4700 uF/16 volt maka gelombang penuh tersebut dibuat menjadi arus listrik searah. Selanjutnya tegangan DC yang keluar dari filter (kondensator) dibuat agar stabil selalu mengeluarkan tegangan sebesar +/- 12 volt (konstan), untuk itu dipasang sebuah IC penstabil tegangan dan dengan memasang IC inilah didapatkan tegangan yang stabil sebesar +/- 12 volt dan dapat menyediakan arus searah sebesar 1 sampai 2 ampere. Oleh karena diperlukan tegangan – 5 volt untuk mendukung operasi RAM dinamik maka pada rangkaian catu daya +/- 12 volt ini perlu ditambahkan sumber tegangan – 5 volt.



Gambar 2.2. Catu Daya +/- 12 volt / 2 ampere dengan penstabil tegangan IC 7812 dan 7912.

Seperti pada catu daya 5 volt, perlu diperhatikan dalam rangkaian ini jangan sampai rangkaian diode jembatan terbalik, pemasangan polaritas kondensator filter terbalik hal ini akan menyebabkan kondensator meletus dan jangan sampai memasang IC penstabil tegangan tertukar kaki-kakinya.

Rangkaian yang ditunjukan pada gambar 2 cukup baik untuk digunakan sebagai pemberi tegangan dan arus listrik pada sistem mikroprosesor, karena berdasarkan hasil pengujian dengan memberikan beban yang bervariasi mulai 0 ampere sampai 2 ampere ternyata tegangan tetap stabil +/- 12 volt.

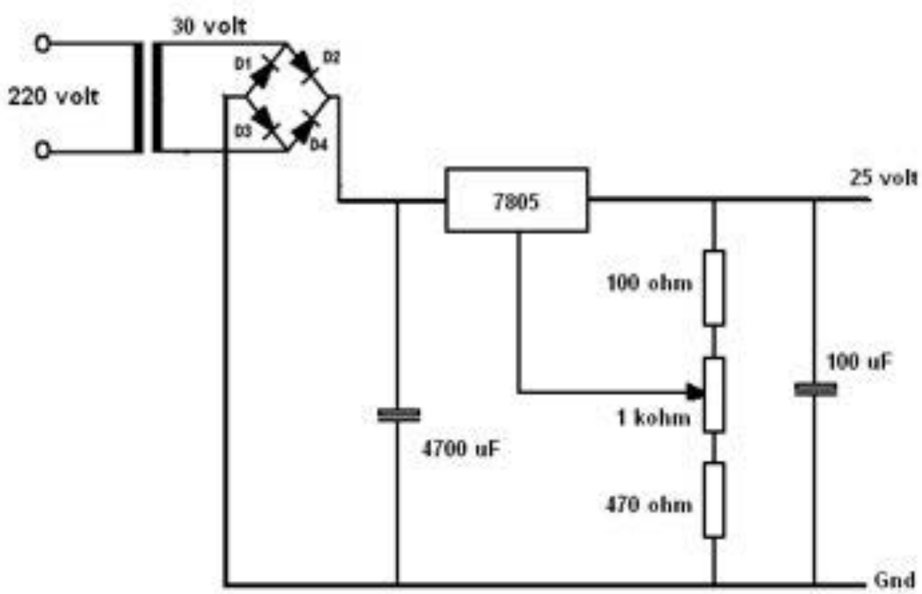
Agar kualitas lebih baik lagi maka IC 7812 dan IC 7912 perlu diberi pendingin dari bahan aluminium, sehingga tidak terjadi lonjakan panas yang tinggi pada IC saat dibebani. Untuk menjaga interferensi frekuensi tinggi maka perlu dipasangkan filter untuk frekuensi tinggi, untuk itu dua kondensator yang masing-masing besarnya 100 nF dipasangkan secara paralel pada kedua kondensator sebelumnya

c. Catu Daya 25 volt.

Catu daya 25 volt difungsikan sebagai pemberi catu tegangan pada rangkaian pemrogram EPROM (2716, 2732 atau 2764), untuk keperluan tersebut rangkaian dapat dibangun dari rangkaian dasar penyearah 4 dioda seperti pada catu daya 5 volt ataupun catu daya +/- 12 volt. Keempat diode membentuk model penyearah gelombang penuh, penyearah tersebut diumpankan pada kumparan sekunder transformator dengan tegangan bolak-balik 30 volt.

Agar tegangan keluaran catu daya dapat memberikan tegangan dapat diatur antara 20 volt sampai 30 volt DC maka diperlukan rangkaian pengatur, dan untuk menjaga stabilitas tegangan yang tinggi maka dipasangkan sebuah rangkaian pengatur tegangan dan sekaligus sebagai penstabil tegangan.

Pada rangkaian catu daya ini IC dengan tipe 7805 berfungsi sebagai pengatur tegangan dan juga sebagai penstabil tegangan, tegangan masukan sebesar 30 volt untuk dapat mencapai tegangan keluaran yang dapat diatur antara 20 volt sampai 30 volt DC sesuai dengan kebutuhan. Untuk menjaga agar IC penstabil tegangan tidak terlalu panas maka tegangan masukan yang berasal dari transformator diambil dengan tegangan sebesar 30 volt untuk keperluan ini dibutuhkan arus listrik sebesar 1 ampere.



Gambar 2.3. Catu daya yang dapat diatur keluarannaya 20 volt.. 30 volt DC

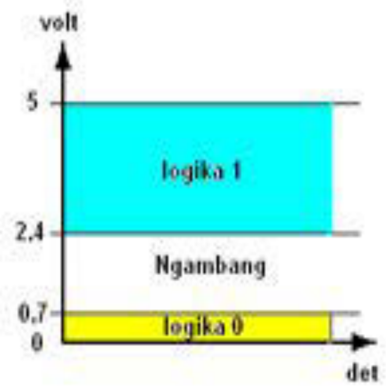
Agar kualitas lebih baik lagi maka IC 7805 perlu diberi pendingin dari bahan aluminium, sehingga tidak terjadi lonjakan panas yang tinggi pada IC saat dibebani. Untuk menjaga interferensi frekuensi tinggi maka perlu dipasangkan filter untuk frekuensi tinggi, untuk itu dua kondensator yang masing-masing besarnya 100 nF dipasangkan secara paralel pada kedua kondensator sebelumnya.

2.1.2. PENSAKLARAN DIGITAL PADA MIKROPROSESOR

Sebagai dasar rangkaian hardware mikroprosesor adalah sistem biner yang hanya mengenal nilai 0 dan 1, atau on dan off sistem saklar elektronik digital. Melalui rangkaian elektronika digital inilah sebuah sistem mikroprosesor dapat dibangun, mulai dari sistem clock, sistem port input/output, sistem register, sistem RAM/ROM dan sistem aritmatik logik unit. Setiap bagian dalam mikroprosesor dibangun dari rangkaian gerbang dasar AND, OR, NOT dan EXOR, dimana setiap gerbang dasar tersebut tersusun dari rangkaian saklar elektronik berupa transistor yang sering disbut dengan Transistor-Transistor-Logic (TTL) atau Dioda-Transistor-Logic (DTL).

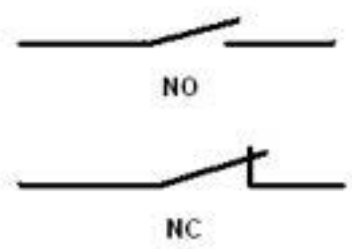
a. Tegangan TTL

Gambar 2.4 merupakan gambar tegangan kerja rangkaian TTL adalah 5 volt, dan untuk status logik 0 memiliki tegangan 0 volt sedangkan status logika 1 memiliki tegangan 5 volt. Walaupun demikian pada kenyataannya nilai tersebut sulit untuk dicapai, oleh karena itu berdasarkan konsensus dunia batas tegangan untuk logika 0 berkisar antara 0 sampai dengan 0,7 volt dan logika 1 berkisar antara 2,4 volt sampai 5 volt. Dengan demikian harus dihindari adanya tegangan antara 0,7 sampai 2,4 volt, pada batas tegangan ini disebut juga dengan status mengambang.



Gambar 2.4. Tegangan Logika

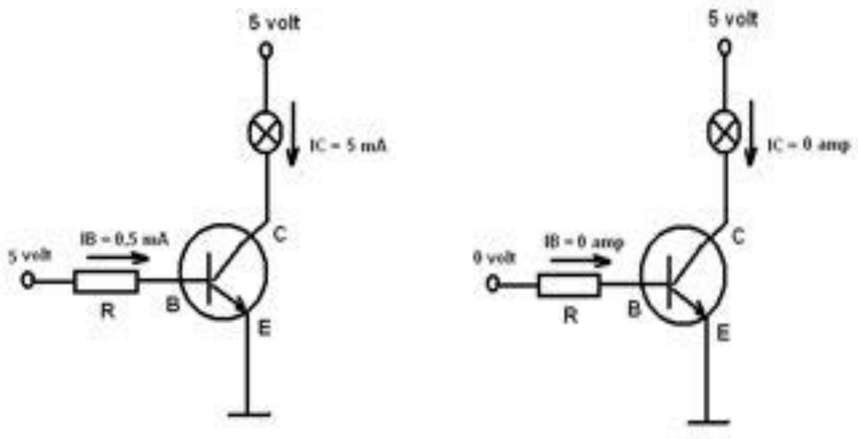
b. Saklar Elektronik



Gambar 2.5. merupakan gambar saklar listrik, yaitu status logika 0 atau logika 1 juga identik dengan sistem saklar listrik yang memiliki kondisi *off* dan kondisi *on*, dan saklar tersebut memiliki 2(dua) kemungkinan kondisi normal-nya yaitu untuk kondisi normal *off* disebut sebagai *normally open* NO dan normal *on* (*normally close)* NC.

Gambar 2.5. Saklar Listrik

Pada saklar listrik berlaku *on* dan *off* dalam kenyataan rangkaian elektronika yang mendukung rangkaian dasar mikroprosessor diwujudkan dalam bentuk rangkaian saklar elektronik, dan setiap saklar elektronika diwakili oleh sebuah transistor. Kondisi on pada saklar diwakili transistor dalam kondisi terhubung penuh antara kolektor dan emitor arus Isistrik mengalir penuh, sedangkan kondisi off pada saklar diwakili transistor dalam kondisi tertutup antara kolektor dan emitor tidak ada arus listrik mengalir sama sekali.



a. Tegangan masukan 0 volt b. Tegangan masukan 5 volt.

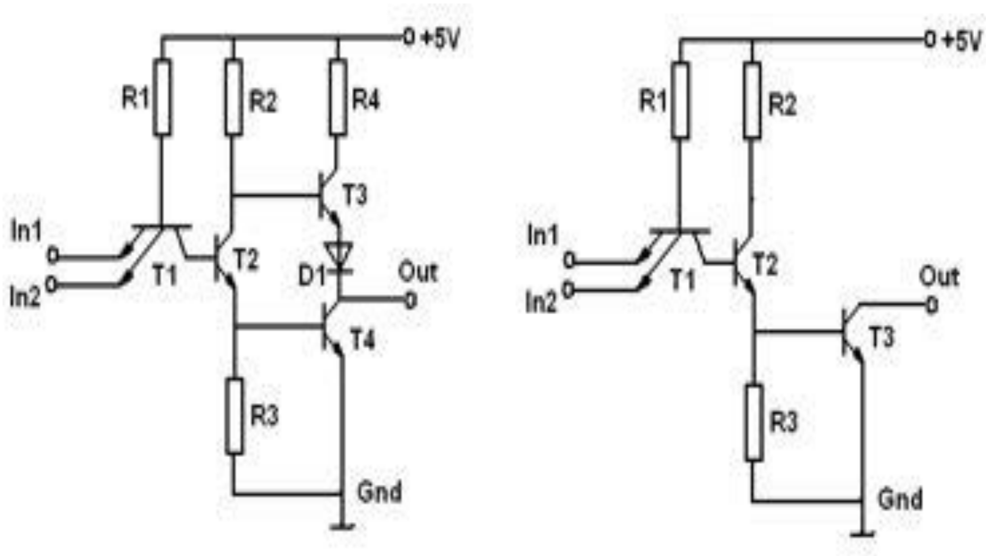
Gambar 2.6. Status *on/off* transistor

Saat tegangan masukan diberikan 0 volt (gambar 6a) artinya terhubung dengan ground, maka tidak ada arus basis yang mengalir ke transistor (*off*) akibatnya resistansi antara kolektor (C) dan emitor (E) bernilai sangat tinggi sehingga tidak ada arus mengalir dari sumber 5 volt melalui beban ke transistor menuju ground (IC=0 amp).

Jika masukan diberikan tegangan 5 volt (gambar 6) maka terjadi aliran arus basis melalui R menuju transistor, selanjutnya pada transistor akan terjadi resistansi kecil antara kolektor dan emitor sehingga IC menjadi 5 mA dan transistor dalam kondisi terhubung (*on*).

c. Rangkaian Elektronik Gerbang Dasar

Melalui rangkaian elektronika yang terdiri dari ribuan transistor, dioda dan komponen pasif inilah tersusun arsitektur gerbang-gerbang digital dasar yang dikenal dengan istilah Transistor-Transistor-Logic (TTL) dan Dioda-Transistor Logic (DTL).



a. TTL keluaran push pull

b. TTL keluaran kolektror terbuka

Gambar 2.7. Gerbang Dasar TTL

Pada gambar 2.7 ditunjukan rangkaian gerbang dasar dengan 2(dua) masukan dan 1(satu) keluaran, untuk gambar 7a rangkaian TTL dengan

keluaran push pull dan gambar 7b rangkaian TTL dengan keluaran kolektror terbuka. Adapun cara kerja rangkaian sebagai berikut:

Untuk gambar 7a berlaku, transistor T1 selalu dicatu tegangan pada kaki basisnya sehingga T1 akan selalu siap dibuat on manakala salah satu In1 atau In2 dihubungkan ke Gnd, dengan demikian akan membuat T2 pada kondisi *on* yang mengakibatkan adanya arus mengalir dari +5 volt melalui R2 ke kolektor T2 menuju emitor T2 dan selanjutnya melewati R3 menuju Gnd. Dengan demikian pada R3 bertegangan positip dan akan memberikan arus basis pada T4, sehingga T4 menjadi on, maka keluaran (out) akan terhubung langsung ke Gnd melalui T4 yang berarti keluaran berlogika 0. Sebaliknya saat In1 dan In2 kondisi terbuka atau dihubungkan ke +5 volt maka T2 akan pada kondisi off, sehingga arus listrik mengalir dari +5 volt melalui R2 menuju ke basis transistor T3 dengan demikian T3 akan on dan dengan kondisi on pada T3 menyebabkan arus listrik dapat mengalir dari +5 volt melalui R4 menuju ke kolektor transistor T3 dan akan diteruskan ke emitor melalui D1 dialirkan ke keluaran (out). Dengan demikian rangkaian pada gambar 7a akan berfungsi sebagai gerbang dasar AND yaitu apabila salah satu atau kedua masukan diberikan logika 0 (0 volt) maka keluaran akan berlogika 0, dan jika kedua masukan diberi logika 1 (+5 volt) maka keluaran akan berlogika 1.

Untuk gambar 7b berlaku sama dengan rangkaian pada gambar b mulai dari T1 dan T2, sedangkan perbedaannya terletak pada keluaran rangkaian. Keluaran pada rangkaian gambar 7b merupakan rangkaian kolektor terbuka (*open collector*), jadi pada saat T3 kondisi on maka keluaran akan terhubung ke Gnd yang berarti berlogika 0 sedangkan saat T3 kondisi off keluaran akan mengambang. Kondisi mengambang tersebut perlu ditentukan logikanya, untuk itu pada keluaran perlu disambung sebuah resistor ke tegangan sumber +5 volt (*pull up resistor*).

2.1.3. GERBANG DIGITAL DASAR

Rangkaian sistem mikrorosessor tidak bisa lepas dari sistem rangkaian digital dasar, karena hampir keseluruhan rangkaian pada sistem mikroprosessor

dibangun dari rangkaian gerbang digital dasar. Aturan main dalam membentuk logika yang diterapkan pada sistem mikroprosessor juga mengikuti aturan logika pada rangkaian gerbang digital dasar, mulai dari sistem bus data, sistem pengalamatan baik memori maupun saluran masukan dan keluaran, sistem register, sistem kendali operasi mikroprosessor dan operasi aritmatik logik unit.

a. Gerbang AND

Gerbang AND minimal memiliki 2(dua) masukan dan 1(satu) keluaran, jika masukan dinyatakan dalam variabel X akan terdapat X_0, X_1. Karena minimal ada 2 masukan maka kemungkinan masukan sampai X_n, sedangkan keluaran hanya terdapat satu keluaran Q.

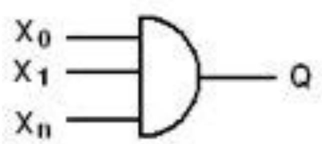
Kondisi logika keluaran Q akan berlogika 1 jika dan hanya jika seluruh masukan X_0, X_1 sampai X_n berlogika 1, jika salah satu masukan berlogika 0 maka keluaran akan berlogika 0. Adapun persamaan untuk gerbang dapat dinyatakan sebagai berikut:

$$Q = X_0 \land X_1 \land \ldots\ldots \land X_n$$

Tabel kebenaran gerbang AND

Masukan		Keluaran
X1	X0	Q
0	0	0
0	1	0
1	0	0
1	1	1

Simbol gerbang AND



b. Gerbang OR

Gerbang OR minimal memiliki 2(dua) masukan dan 1(satu) keluaran, jika masukan dinyatakan dalam variabel X akan terdapat X_0, X_1. Karena minimal ada 2 masukan maka kemungkinan masukan sampai X_n, sedangkan keluaran hanya terdapat satu keluaran Q.

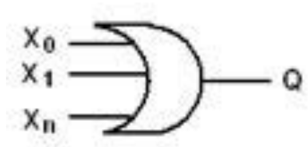
Kondisi logika keluaran Q akan berlogika 0 jika dan hanya jika seluruh masukan X_0, X_1 sampai X_n berlogika 0, jika salah satu masukan berlogika 1 maka keluaran akan berlogika 1. Adapun persamaan untuk gerbang dapat dinyatakan sebagai berikut:

$$\mathbf{Q = X_0 \vee X_1 \vee \ldots\ldots \vee X_n}$$

Tabel kebenaran gerbang OR

Masukan		Keluaran
X1	X0	Q
0	0	0
0	1	1
1	0	1
1	1	1

Simbol gerbang OR



c. Gerbang XOR

Berikut merupakan gerbang XOR dengan 2(dua) masukan dinyatakan dalam variabel X_0, X_1. dan 1(satu) keluaran dinyatakan dalam variabel Q.

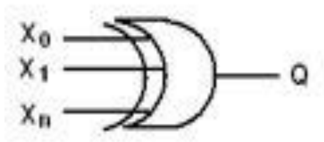
Kondisi logika keluaran Q akan berlogika 0 jika dan hanya jika seluruh masukan X_0, X_1 memiliki logika yang sama yaitu berlogika 0 atau 1, jika skedua masukan memiliki logika tidak sama maka keluaran akan berlogika 1. Adapun persamaan untuk gerbang dapat dinyatakan sebagai berikut:

$$\mathbf{Q = X_0 \veebar X_1}$$

Tabel kebenaran gerbang XOR

Simbol gerbang XOR

Masukan		Keluaran
X1	X0	Q
0	0	0
0	1	1
1	0	1
1	1	0



d. Gerbang NOT

Pada gerbang NOT terdapat hanya satu masukan dan satu keluaran, jika dinyatakan mariabel masukan X dankeluaran Q maka logika masukan X akan selalu kebalikan dari logika keluaran.

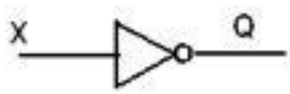
Sebagai contoh jika X berlogika 1 maka Q akan berlogika 0, begitu juga jika X berlogika 0 maka Q akan berlogika 1. Adapun persamaan untuk gerbang dapat dinyatakan sebagai berikut:

$$Q = \bar{X}$$

Tabel kebenaran gerbang NOT

Masukan	Keluaran
X	Q
0	1
1	0

Simbol gerbang NOT



e. Gerbang NAND, NOR

Kedua gerbang ini pada prinsipnya merupakan gabungan dari dua gerbang dasar yang salah satunya adalah gerbang NOT, sehingga untuk gerbang AND jika digabung dengan NOT akan menjadi gerbang baru yaitu NAND dan untuk gerbang OR digabung NOT menjadi gerbang NOR.

Tabel kebenaran untuk logika keluaran pada gerbang NAND sama dengan gerbang AND akan tetapi dibalikan, dan untuk keluaran gerbang NOR sama dengan gerbang OR akan tetapi juga dibalikan. Adapun persamaan untuk gerbang dapat dinyatakan sebagai berikut:

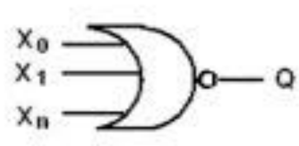
- Untuk gerbang NOR berlaku:

$$Q = \overline{X0 \vee X1 \vee \ldots\ldots \vee Xn}$$

Tabel kebenaran gerbang NOR

Masukan		Keluaran
X1	X0	Q
0	0	0
0	1	1
1	0	1
1	1	1

Simbol gerbang NOR



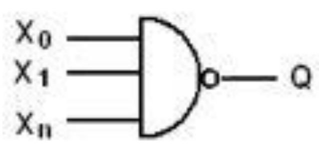
- Untuk gerbang NAND berlaku:

$$Q = \overline{X0 \wedge X1 \wedge \ldots\ldots \wedge Xn}$$

Tabel kebenaran gerbang NAND

Masukan		Keluaran
X1	X0	Q
0	0	1
0	1	1
1	0	1
1	1	0

Simbol gerbang NAND



2.1.4. RANGKAIAN DIGITAL KOMPONEN MIKROPROSESSOR

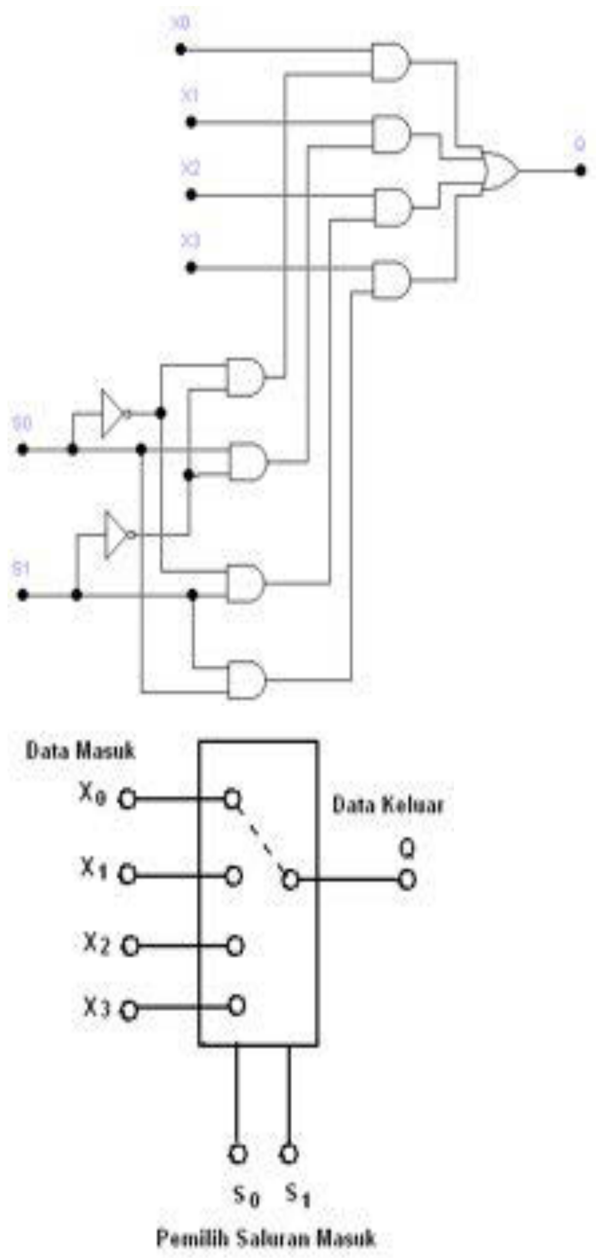
Setiap bagian dalam sistem mikroprosessor dibentuk dari rangkaian digital, jadi berdasarkan fungsi gerbang dasar AND, OR, NOT, EXOR, NAND dan NOR dapat dimanfaatkan untuk membentuk rangkaian sesuai dengan kebutuhan pada sistem.

Berikut dijelaskan rangkaian dasar pembentuk bagian yang ada pada sistem mikroprosessor yang dibangun dari gerbang dasar:

a. Pemilih Data (Data Selector)

Pada sistem bus baik untuk data maupun penunjuk alamat memori atau penunjuk alamat port I/O pada sistem mikroprosessor diperlukan pemilih data, sebagai contoh pada saat memasukan data dari memori pada alamat tertentu ke mikro data dilewatkan bus data dan pada saat bersamaan data dari memori pada alamat lain tidak boleh masuk. Dengan demikian harus dilakukan pemilihan data berasal dari mana yang boleh masuk, untuk itulah diperlukan sebuah pemilih data.

Pad gambar 8 ditunjukan pemilih data 4 ke 1, artinya terdapat 4(empat) buah saluran data masuk dan hanya ada satu saluran data keluar. Data masuk melalui saluran X0, X1, X2 dan X3, sedangkan saluran keluar melalui Q, untuk memilih saluran mana yang boleh masuk dilakukan dengan memberikan kombinasi logik pada pemilih S0 dan S1.



Gambar 2.8. Rangkaian Pemilih Data.

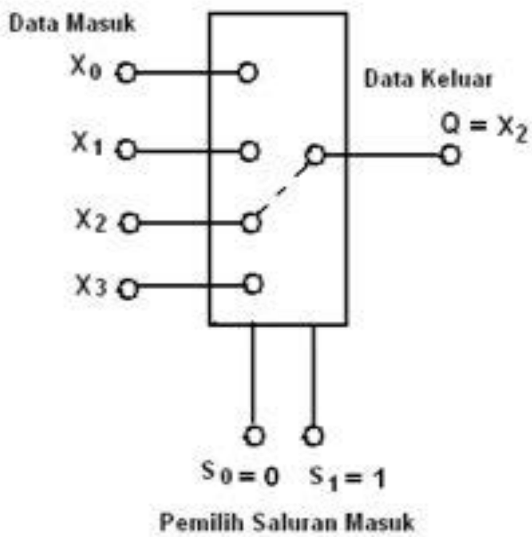
Pemilihan data dilakukan melalui kombinsasi S0 dn S1, sehingga data X mana yang bisa masuk pemilih data untuk bisa dikeluarkan melalui keluaran Q, untuk itu dapat dilihat pada tabel fungsi berikut:

Tabel Pemilih data masuk melalui kombinasi S_0 dan S_1

Masukan		Keluaran
S_1	S_0	Q
0	0	X_0
0	1	X_1
1	0	X_2
1	1	X_3

Contoh:

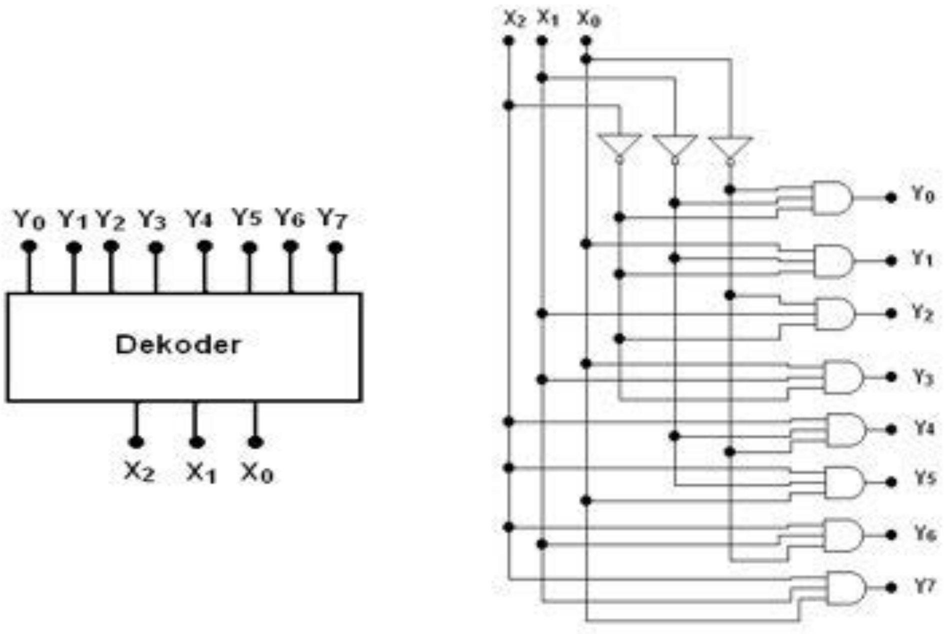
Pemilih data seperti pada gambar 8 dengan X_0 merupakan saluran data dari memori A, X_1 dari memori B, X_2 dari memori C dan X_3 saluran dari memori D. Jika S_0 dan S_1 merupakan saluran penunjuk alamat memori mendapatkan kondisi logika S_0 = 0 dan S_1 =1, maka akan dipilih data yang berasal dari memori C dan saluran yang digunakan adalah X_2.



Gambar 2.9. Hasil pemilihan saluran data berdasarkan S_0 dan S_1

b. Dekoder

Dekoder banyak digunakan dalam sistem mikroprosessor untuk menunjuk alamat lokasi memori atau alamat lokasi port I/O, kombinasi logik saluran masukan dekoder berfungsi sebagai penentu saluran mana yang dipilih. Untuk menunjang pemakaian tersebut telah banyak dibuat dalam bentuk rangkaian digital yang sudah terintegrasi dalam bentuk IC (74138, 74139), sehingga pengembang sistem mikroprosessor tidak perlu lagi erpikir tentang rangkaian digital di dalamnya.



a. Diagram Blok Dekoder b. Rangkaian Dekoder

Gambar 2.10. Dekoder 3 ke 8

Pada tahap kali ini disajikan rangkaian digital dari sebuah dekoder 3 masukan untuk mengaktifkan 8 keluaran, dengan tujuan untuk memberikan gambaran fungsi dekoder 3 ke 8. Gambar 10 dapat memberikan penjelasan berdasarkan rangkaian logika tentang fungsi sebuah dekoder, dengan memberikan kombinasi logika pada masukan X_0, X_1 dan X_2 maka dapat mengaktifkan salah satu keluaran Y_0, Y_1,…….Y_7. Melalui tabel fungsi dekoder berikut akan lebih jelas diskrpsi fungsi dekoder, logik aktif untuk keluaran Y_0, Y_1,…….Y_7 adalah logika 1.

Tabel Fungsi Dekoder

X_2	X_1	X_0	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7

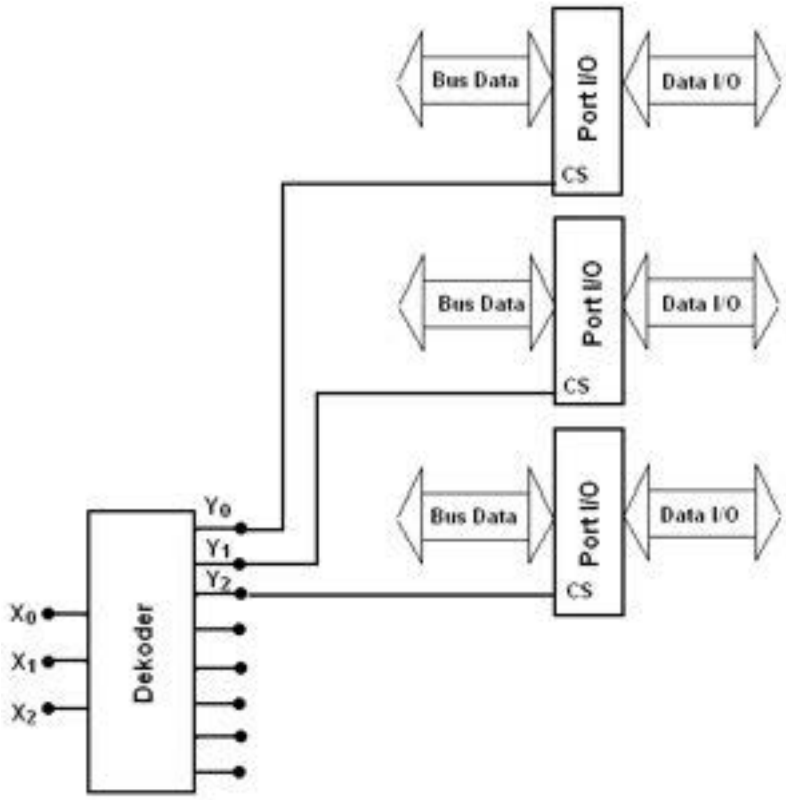


0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0		1

Contoh:

Dalam sebuah sistem mikroprosessor dikembangkan 3(tiga) buah port I/O yang dimanfaatkan untuk membaca dan menuliskan data, setiap port I/O diwakili sebuah chip IC yang dilengkapi dengan chip select (CS) aktif high. Dengan memanfaatkan sebuah dekoder maka penunjukan IC mana yang harus aktif dapat dilakukan, berdasarkan fungsi dekoder 3 ke 8 maka kombinasi X_0, X_1 dan X_2 dapat digunakan untuk menunjuk port I/O mana yang aktif.

Gambar 2.11 menunjukan rangkaian pemilih port I/O menggunakan dekoder, kombinasi X_0, X_1 dan X_2 menentukan port mana yang digunakan untuk menghubungkan antara Data I/O dengan Bus Data. Jika $X_0 = 0$, $X_1 = 0$ dan $X_2 =0$ maka $Y_0 = 1$ yang berarti Port I/O pertama yang aktif, jika $X_0 = 1$, $X_1 = 0$ dan $X_2 =0$ maka $Y_1 = 1$ yang berarti Port I/O ke dua yang aktif dan jika $X_0 = 0$, $X_1 = 1$ dan $X_2 =0$ maka $Y_2 = 1$ yang berarti Port I/O ke tiga yang aktif. Oleh karena keluaran dekoder yang lain tidak digunakan (terbuka) maka untuk kombinasi X_0, X_1 dan X_2 yang lain tidak berdampak pada sistem rangkaian.



Gambar 2.11. Aplikasi Dekoder Sebagai penunjuk Port I/O

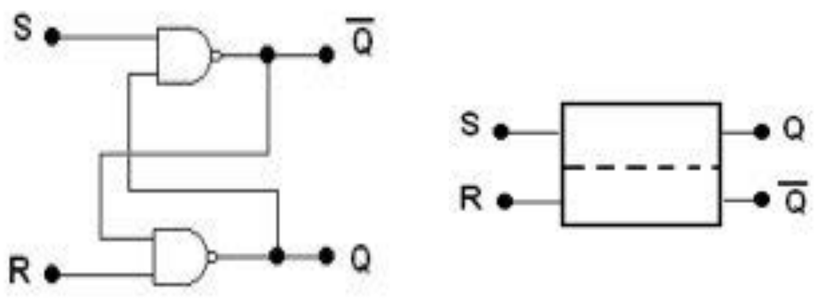
c. Penyimpan 1 Bit Data

Data dalam sistem mikroprosessor menempati tempat yang paling penting, karena data tersebutlah sebagai bahan dasar yang diolah melalui proses aritmatika dan logik. Data pada sistem mikroprosessor dikemas dalam bentuk byte atau 8 bit, sedangkan bit merupakan satuan terkecil dari data digital. Dengan demikian bit data harus difasilitasi dengan sistem penyimpanan, untuk menyimpan 1(satu) bit data dibutuhkan 1 bit memori yang dapat dibangun dari sebuah flip-flop.

- **RS-Flip-flop**

Gambar 12a merupakan sebuah flip-flop sebagai penyimpan 1 bit data yang dibangun dari gerbang dasar NAND dan secara diagram blok ditunjukan gambar 12 b, flip-flop disamping dibangun dari gerbang dasr NAND dapat juga dibangun dari gerbang dasar NOR. Pada flip-flop dikenal dengan istilah

set (S) dan reset (R) serta terdapat 2(dua) buah keluaran yang keduanya selalu berlogika kebalikan yaitu Q dan $\overline{Q}$. Berikut merupakan rangkaian gerbang dasar flip-flop menggunakan gerbang dasar NAND,



a. Rangkaian Dasar RS-FF b. Diagram Blok RS-FF

Gambar 2.12 RS - Flip-Flop

Jika S dan R masing-masing diberi logika 1 maka tidak ada perubahan pada keluaran Q dan $\overline{Q}$, jika S = 0 dan R = 1 maka flip-flop di-reset dan jika S = 1 dan R = 0 maka flip-flop di-set. Sedangkan untuk kondisi S = 0 dan R = 0 tidak perkenankan karena kedua keluaran akan berlogika 1, artinya rangkaian tidak berfungsi sebagai flip-flop. Tabel fungsi berikut merupakan kondisi logika pada sebuah RS-FF:

Tabel Fungsi RS-Flip-flop

S	R	Q	$\overline{Q}$
0	0	X	X
0	1	0	1
1	0	1	0
1	1	Q	$\overline{Q}$

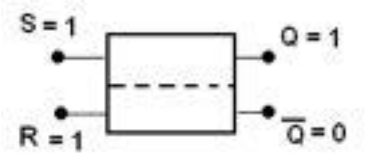
Sebuah RS-FF untuk mempertahankan data yang tersimpan di dalamnya yaitu pada keluaran Q dan $\overline{Q}$, kondisi S harus berada pada kondisi logika 1 dan kondisi R juga harus dipertahankan pada kondisi logika 1. Pada kondisi tersebut RS-FF berfungsi sebagai penyimpan bit data sebagai hasil proses set atau reset sebelumnya, dan jika ingin menyimpan bit data dengan logika 0

yang berarti Q diisi data dengan logika 0 maka RS-FF harus di-reset dan jika diisi data dengan logika 1 maka harus dilakukan set.

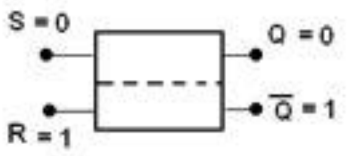
Contoh:

> Kondisi RS-FF sebelumnya Q = 1 dan $\overline{Q}$ = 0, pada kondisi ini S = 1 dan R = 1 berarti RS-FF sedang menyimpan data berlogika 1. Kemudian pada S diberikan logika 0 sesaat setelah terjadi perubahan pada keluaran RS-FF logika S diubah kembali ke logika 1, dengan demikian setelah terjadi reset akan terjadi perubahan pada keluaran Q dan dipertahankan data tersebut sampai proses berikutnya.

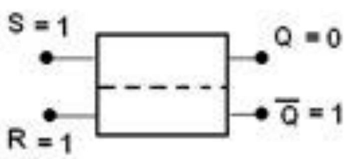
Kondisi awal RS-FF



Kondisi RS-FF di-reset



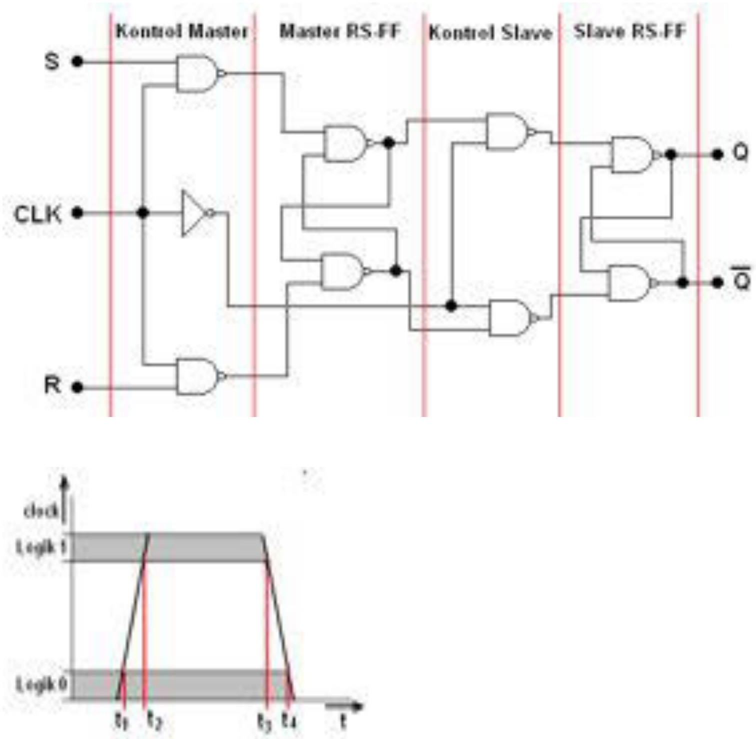
Kondisi akhir setelah di-reset



- ***Master-Slave* Flip=flop**

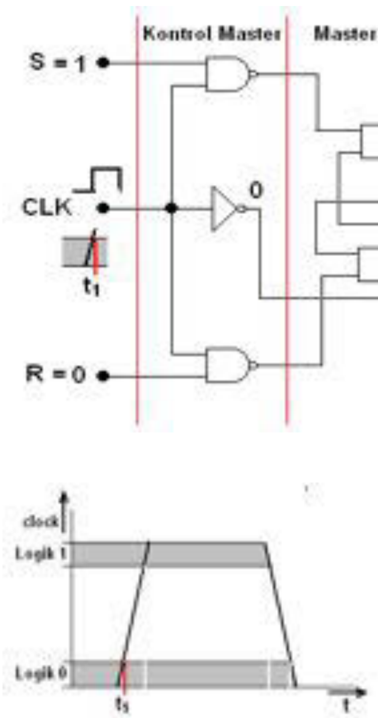
Gambar 2.13 merupakan sebuah *master-slave* flip-flop sebagai penyimpan 1 bit data, rangkaian ini adalah rangkaian pengembangan RS-FF yang dibangun

dari gerbang dasar NAND. Secara prinsip mempunyai fungsi yang sama dengan RS-FF, perbedaannya pada proses set maupun reset dilakukan melalui pemberian *clock* dan terdapat 2(dua) RS-FF di dalamnya yang masing-masing berfungsi sebagai *master* dan berfungsi sebagai *slave*.



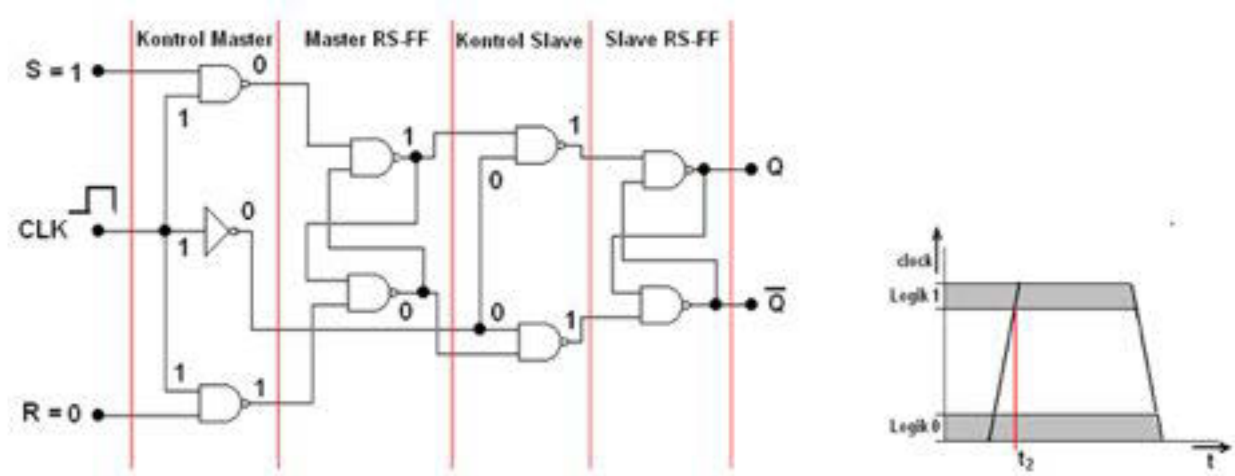
Gambar 2.13. Rangkaian *Master-Slave* RS-FF

Rangkaian *Master-Slave* RS-FF ini prinsip kerjanya sangat tergantung dari sistem *clock* yang diberikan, dan *clock* yang diberikan berfungsi untuk mengendalikan proses set ataupun proses reset. Saat dilakukan *set* maka kontrol master berfungsi melalui clock *set* dilakukan pada master, dilanjutkan dengan set pada master kemudian kontrol slave bekerja dan terjadi proses set pada slave. Demikian juga pada saat dilakukan reset akan diawali dari kontrol master dan dilanjutkan penulisan keluaran *master* ke *slave*, kontrol memasukan data ke *master* dan dilanjutkan penulisan ke *slave* dilakukan oleh sinyal satu kali *clock* yaitu t_1 sampai dengan t_4. Secara rinci proses set dan reset master-slave RS-FF yang dikendalikan oleh clock sebagai berikut:



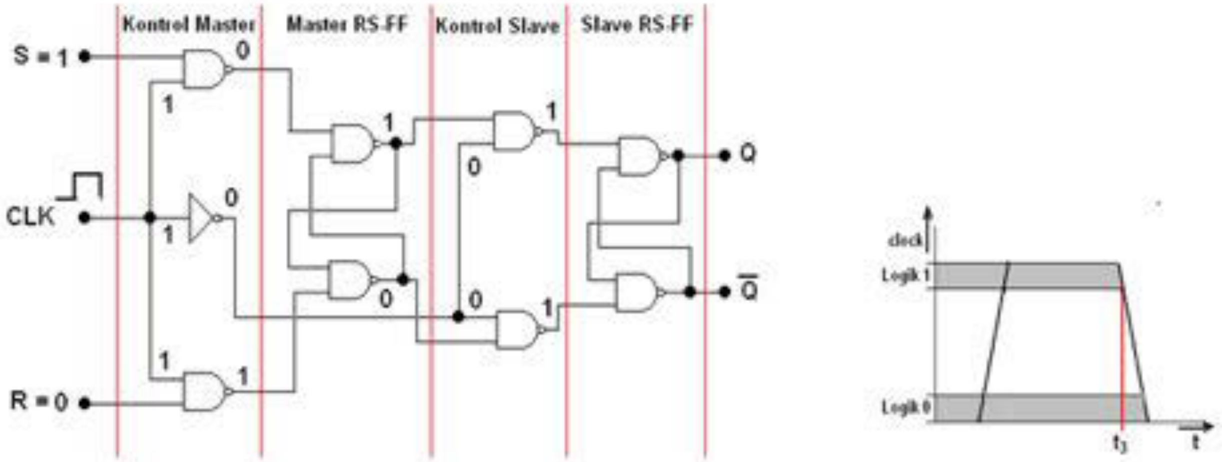
Gambar 2.14. Kondisi awal proses *set* pada *master-slave* RS-FF

t_1 merupakan awal *clock* yaitu kondisi akhir logik 0 pada *clock* menuju logik 1, saat ini keluaran gerbang NOT pada clock dipertahankan pada logika 0 sehingga masukan pada *slave* RS-FF berlogika 1dan *slave* RS-FF tetap mempertahankan kondisi logik sebelumnya.



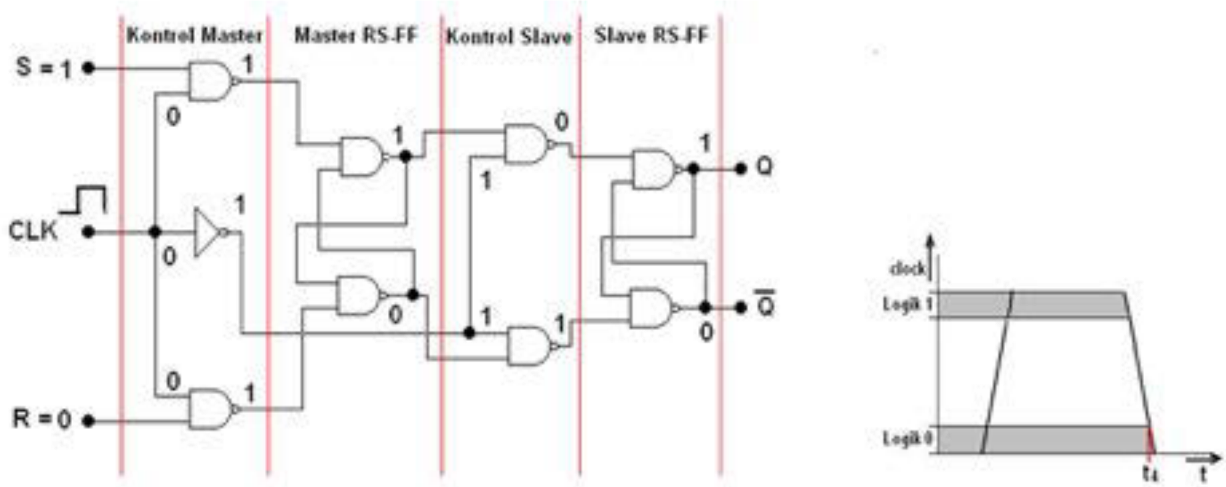
Gambar 2.15. Kondisi *clock* pada t_2 proses *set* pada *master-slave* RS-FF

t_2 merupakan batas awal logika 1 yang dicapai *clock* naik dari logik 0, saat ini masukan master terhubung. Dengan demikian keluaran master RS-FF ditentukan logikanya melalui masukan S dan R, berdasarkan logika pada gambar 2.15 maka saat t_2 keluaran kontrol berlogika 1 dan keluaran slave RS-FF tetap memegang bit data sebeliumnya.



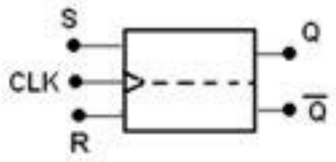
Gambar 2.16. Kondisi *clock* pada t_3 proses *set* pada *master-slave* RS-FF

Gambar 2.16 menunjukan kondisi RS-FF pada saat t_3 merupakan batas akhir logika 1 yang dicapai *clock*, saat ini clock mulai turun menuju logika 0. Kondisi keluaran master RS-FF maupun keluaran slave RS-FF masih dipegang, yaitu sama dengan logika sebelumnya.



Gambar 2.17. Kondisi *clock* pada t_4 proses *set* pada *master-slave* RS-FF

Melalui sistem *clock* inilah semua rangkaian digital dalam sistem mikroprosessor dioperasikan, sehingga proses pemberian *clock* merupakan hal yang sangat penting dan secara prinsip kerja *clock* untuk setiap bagian sama. Simbol dalam diagram blok untuk *master-slave* RS-FF sebagainberikut:

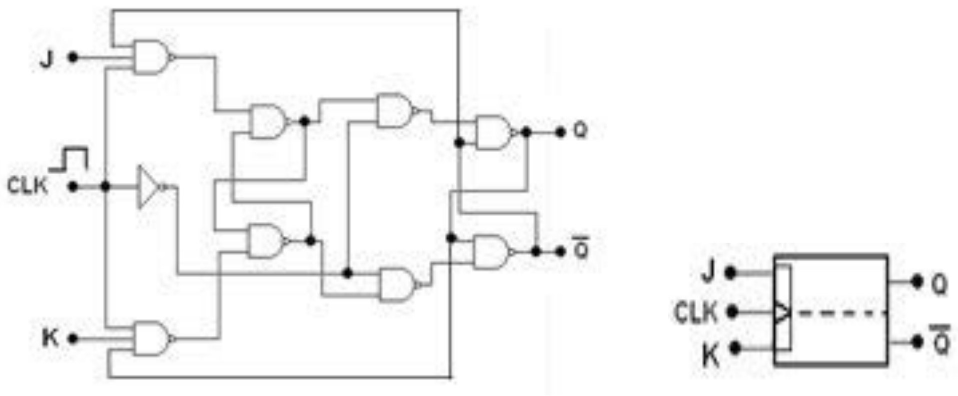


Gambar 2.17 menunjukan kondisi RS-FF pada saat t_4 merupakan batas awal logika 0 yang dicapai *clock*, saat ini *clock* mulai masuk daerah logika 0. Kondisi keluaran *master* RS-FF masih dipegang dan keluaran *slave* RS-FF di-set dari logika sebelumnya menjadi logika 1. Dengan demikian kondisi keluaran *master* dan kondisi keluaran *slave* berlogika sama, yaitu masing-masing sudah selesai di-set.

- ***Master-Slave* JK-FF**

Master-slave JK-FF memiliki rangkaian hampir sama dengan rangkaian yang dimiliki oleh master-slave RS-FF, perbedaannya hanya pada master-slave JK-FF terdapat rangkaian balik (*feedback*) dari keluaran menuju ke gerbang kontrol master.

Jika pada RS-FF terdapat kombinasi masukan (S=0, R=0) yang tidak diijinkan dikarenakan keluaran tidak bisa digunakan sebagai fungsi flip-flop, maka pada JK-FF ini semua kombinasi masukan (J dan K) dapat dimanfaatkan semuannya. Gambar 18 merupakan rangkaian dasar dari sebuah master-slave JK-FF, keluaran Q diumpan balikan ke gerbang kontrol master terkoneksi dengan masukan J dan untuk $\bar{Q}$ diumpan balikan ke gerbang masukan K.



a. Rangkaian gerbang dasar JK-FF b. Diagram blok JK-FF

Gambar 2.18. Master-Slave JK-FF

Jika diberikan data pada masing-masing masukan J =0 dan K =0 pada pemberian clock waktu ke t_n, maka keluaran JK-FF tidak akan mengalami perubahan artinya tetap memegang data sebelumnya (fungsi memori) pada clock waktu ke t_{n+1}.

Jika diberikan data pada masing-masing masukan J =1 dan K =0 pada pemberian clock waktu ke t_n, maka keluaran JK-FF akan mengalami perubahan pada *clock* waktu ke t_{n+1}. Pada keluaran terjadi perubahan dari kondisi semula menjadi Q = 1 dan $\bar{Q}$ = 0, begitu juga sebaliknya jika J = 0 dan K =1 maka setelah *clock* waktu ke t_{n+1} terjadi perubahan dari kondisi semula menjadi Q = 0 dan $\bar{Q}$ = 1.

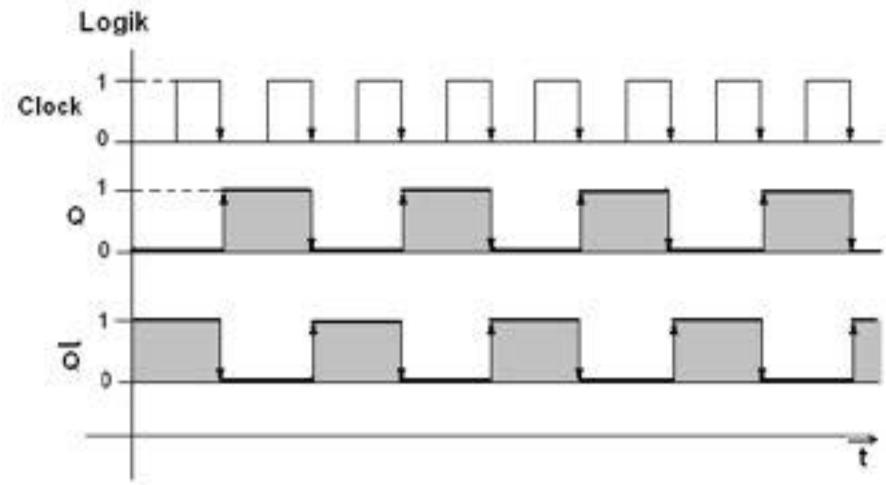
Jika diberikan data pada masing-masing masukan J =1 dan K = 1 pada pemberian clock waktu ke t_n, maka keluaran JK-FF akan mengalami perubahan pada *clock* waktu ke t_{n+1}. Pada keluaran terjadi perubahan dari kondisi semula menjadi Q = $\bar{Q}$ dan $\bar{Q}$ = Q, begitu juga setelah *clock* waktu ke t_{n+2} terjadi perubahan kembali Q = $\bar{Q}$ dan $\bar{Q}$ = Q, kejadian ini akan berlangsung terus untuk *clock* berikutnya dan kejadian ini disebut dengan *toggle*.

Berdasarkan perubahan kondisi keluaran yang disebabkan pemberian variasi kombinasi logik pada J dan K serta pemberian clock pada *master-slave* JK-FF, maka tabel berikut merupakan fungsi dari kondisi tersebut:

Tabel Fungsi *master-slave* JK-FF

tn		tn+1		
K	J	Q	$\bar{Q}$	
0	0	Q	$\bar{Q}$	fungsi memori
0	1	1	0	
1	0	0	1	
1	1	$\bar{Q}$	Q	fungsi toggle

Gambar 2.19 menunjukan diagram impul pada *master-slave* JK-FF saat masukan masing-masing diberi logika J = 1 dan K =1 dan *clock* diberikan secara berturutan t_1.......t_n, ternyata pada setiap selesai *clock* terjadi *toggle* antara Q dengan $\bar{Q}$ seperti digambarkan berikut:



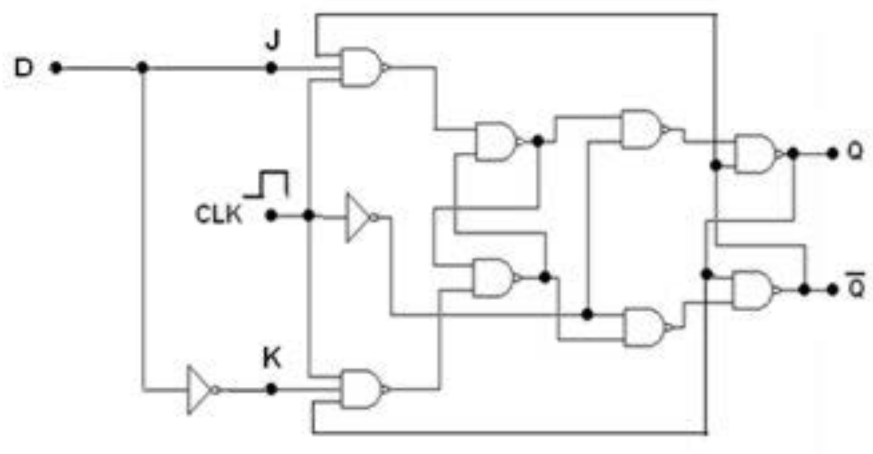
Gambar 2.19. Diagram impul pada master-slave JK-FF

- **D-Flip-Flop**

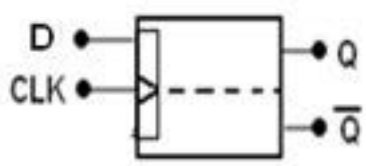
D-FF adalah sebuah penyimpan bit data yang hanya memiliki satu masukan dan dua keluaran, untuk memindahkan bit data yang diberikan pada masukan dilakukan dengan memberikan sebuah *clock*.

Gambar 2.20 merupakan prinsip rangkaian D-FF yang dibangun dari sebuah JK-FF, yaitu dengan menambahkan sebuah gerbang dasar NOT pada masukan K dan masukan pada gerbang dasar NOT disambungkan dengan masukan J pada JK-FF.

Huruf D pada D-FF merupakan singkatan dari kata *delay* yang artinya tunda waktu, perubahan pada keluaran Q dan $\overline{Q}$ terjadi setelah terjadi clock dan logika Q akan selalu sama dengan D.



a. Rangkaian D-FF dibangun dari JK-FF



b. diagram blok D-FF

Gambar 2.20. Rangkaian dan Diagram Blok D Flip-flop

Masukan JK-FF dibuat berlogika komplemen antara masukan J dan masukan K pada JK-flip-flop (diberi gerbang NOT), berdasarkan tabel JK-FF setelah diberi satu kali *clock* diperolah logika keluaran Q sama dengan logika masukan J. Selanjutnya masukan J tersebut diberiri notasi D yang artinya D-flip-flop, D-FF banyak digunakan untuk menyimpan bit data seperti register,

memori atau port keluaran. Sehingga fungsi D-flip-flop dapat dibuat dalam bentuk tabel fungsi seperti ditunjukan pada tabel fungsi berikut:

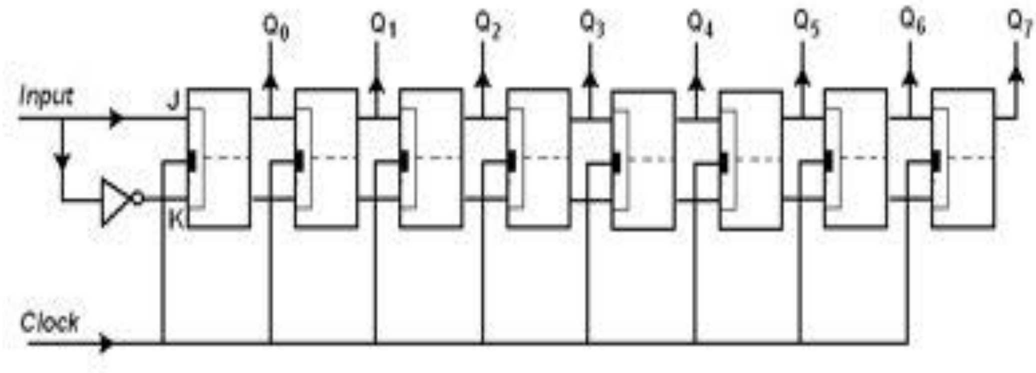
Tabel Fungsi D-FF

tn	tn+1	
D	Q	$\overline{Q}$
0	0	1
1	1	0

d. Penyimpan 8 Bit Data

Dalam sistem miroprosesor data yang digunakan untuk dimanipulasi atau diproses baik dalam aritmatika atau logika adalah 8 bit data, adapun 8 bit data tersebut merupakan satu kesatuan artinya bit yang satu dengan bit lainnya hanya mempunyai nilai apabila menjadi satu kesatuan yang disebut dengan byte.

Sebagai penyimpan 1 bit data telah dikenalkan dengan nama flip-flop yang terdiri dari beberapa tipe flip-flop, maka untuk penyimpan 1 byte atau sama dengan 8 bit data digunakan sejumlah 8 buah flip-flop yang disusun sedemikian rupa sehingga membentuk sebuah rangkaian penyimpan 8 bit dengan nama register.



Gambar 2.21. Penyimpan Data 8 Bit (Register 8 Bit)

Gambar 2.21 menunjukan rangkaian penyimpan data 8 bit yang dibangun dari 8 buah JK-FF, seperti diketahui bahwa JK-FF jika kedua masukannya dibuat berlawanan yaitu untuk J =1 maka K=0 dan untuk J=0 maka K=1 ternyata keluaran Q merupakan togle dari keluaran Q sebelumnya. Dengan demikian jika dirangkai akan menjadi sebuah register 8 bit yang mampu menyimpan data 8 bit (1byte). Sedangkan proses penyimpanan data diawali dari pemberian data 8 bit pada saluran input dan secara berturut-turut diberi sejumlah 8 clock, maka data akan tersimpan pada Q_0 sampai Q_7.

Hampir setiap operasi di dalam sistem mikroprosessor selalu menggunakan register, sehingga register dalam aplikasinya disamping berfungsi sebagai penyimpan 8 bit dalam sistem mikroprosessor juga berfungsi sebagai:

a. Register Geser (Shift Registera)
b. Register Masuk Serial Keluar Serial (Serial In Serial Out / SISO)
c. Register Masuk Serial Keluar Paralel (Serial In Paralel Out / SIPO)
d. Register Masuk Paralel Keluar Serial (Paralel In Serial Out / PISO)
e. Register Masuk Paralel Keluar Paralel (Paralel In Paralel Out / PIPO)

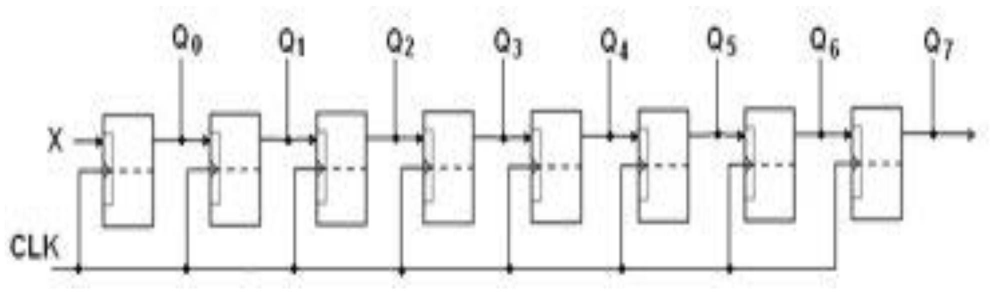
- **Register Geser (Shift Register)**

Register geser merupakan register yang datanya bisa digeser secara serial, melalui pemberian *clock* data yang berada di flip-flop sebelumnya dipindahkan flip-flop selanjutnya. Gambar 2.23 menunjukan data masukan pada register secara serial melalui X dan dengan bantuan sistem *clock* data digeser ke Q_0, Q_1, Q_2 sampai ke Q_7.

Jika data dikeluarkan secara serial melalui Q7, yaitu dengan memberikan clock diberikan sampai data digeser berada pada Q7 maka data dapat dibaca secara paralel melalui Q_0, Q_1, Q_2 sampai ke Q_7.

Jika data yang digeser sudah dapat dibaca secara paralel melalui Q_0, Q_1, Q_2 sampai ke Q_7 maka data dapat dibaca berdasarkan tingkatannya, yaitu $Q_0 = 2^0$, $Q_1=2^1$, $Q_2=2^2$ sampai $Q_7=2^7$. Jadi data paralel pada register merupakan data yang nilainya 0 sampai dengan 255.

Pada dasarnya merupakan koneksi seri dari Flip flop seperti gambar berikut:

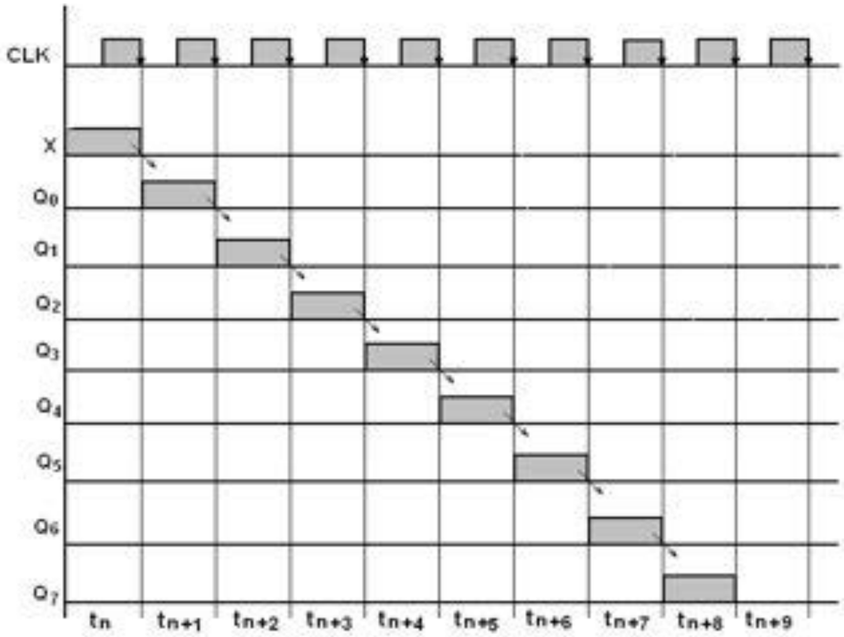


Gambar 2.22. Rangkaian Register Geser 8 Bit

Adapun prinsip kerja register geser pada gambar 2.22 adalah sebagi berikut:

Dengan menempatkan data masukan X dengan logika 1 selama Clock pertama (t_n) dan diberikan logika 0 sampai clock ke t_{n+9}, sementara itu semua keluaran dari Q_0 sampai dengan Q_7 pada clock pertama berlogika 0, maka melalui pemberian *clock* pertama (t_n) sampai *clock* ke t_{n+9} data yang berada pada masukan X akan diterima petama kali oleh Q_0, kemudian digeser ke Q_1 sehingga Q_0=0 dan Q_1=1 sementara itu Q_2 sampai Q_7 tetap sama dengan 0.

Setelah *clock* berikutnya yaitu t_{n+2} maka Q_0=0 dan Q_1=0 dan Q_2=1 sementara itu Q_3 sampai Q_7 tetap sama dengan 0. Setelah *clock* t_{n+3} maka Q_0=0, Q_1=0, Q_2=0 dan Q_3=1 sementara itu Q_4 sampai Q_7 tetap sama dengan 0. Setelah *clock* t_{n+3} maka Q_0=0 sampai Q_3=0 dan Q_4=1 sementara itu Q_5 sampai Q_7 tetap sama dengan 0. Demikian seterusnya, setelah diberikan *clock* t_{n+7} maka Q_0=0 sampai Q_6=0 dan Q_7=1, jika diberikan *clock* berikutnya yaitu t_{n+8} maka Q_0=0 sampai Q_7=0. Prinsip kerja register geser di atas secara urutan diagram pulsa ditunjukan pada gambar 2.23.



Gambar 2.23. diagram pulsa register geser 8 bit

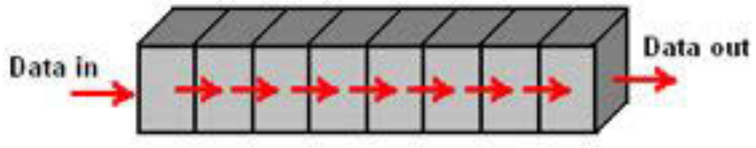
Register geser (*Shift register*) 8 bit (1byte) dapat digunakan sebagai penyimpanan data yang akan diproses atau sebagai penyimpan hasil proses mikroprosessor, sifat data yang tersimpan adalah sementara jika sumber catu daya tidak ada maka hilang pula informasi yang tersimpan di dalamnya.

Terdapat 4(empat) fungsi dari rangkaian dasar register, yaitu meliputi:

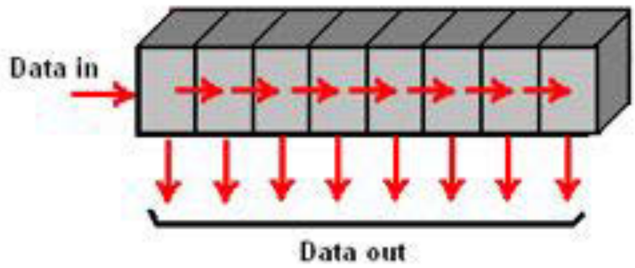
- Register Masuk Serial Keluar Paralel (Serial In Paralel Out / SIPO)
- Register Masuk Serial Keluar Serial (Serial In Serial Out / SISO).
- Register Masuk Paralel Keluar Serial (Paralel In Serial Out / PISO)
- Register Masuk Paralel Keluar Paralel (Paralel In Paralel Out / PIPO)

Penerapan keempat fungsi rangkaian pada operasi dasar register dalam sistem mikroprosesor ditunjukan pada gambar 2.24, yaitu rangkaian register dengan arsitekturnya disamping diguna-kan sebagai register geser dapat digunakan sebagai rangkaian dasar untuk beberapa operasi, seperti SISO, SIPO, PISO,PIPO, geser kiri dan geser kanan. Sebagai contoh untuk geser kanan dan bit akhir dihubungkan ke data in, maka dengan komposisi bit data $(0110\ 1110)_B = (110)_D$ digeser ke kanan 1 bit data yang tersimpan dalam register adalah $(0011\ 0111)_B = (55)_D$. Jika hasil tersebut digeser ke kiri dua bit maka hasilnya adalah $(1101\ 1100)_B = (220)_D$, dengan demikian sangat

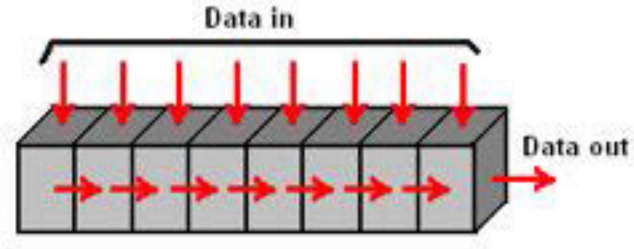
mungkin melakukan manipulasi bit data melalui operasi register dalam sebuah mikroprosesor.



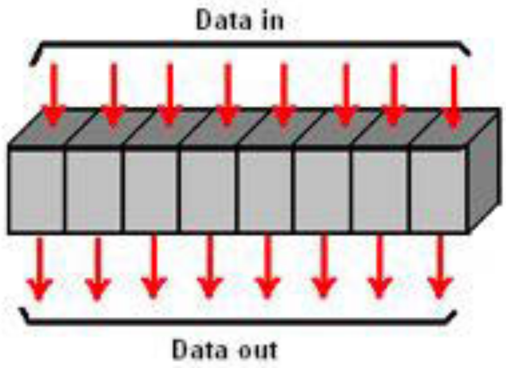
- Operasi geser ke kanan register 8 bit



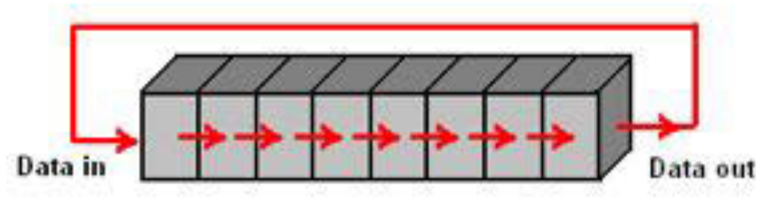
- Operasi serial input dan parallel output register 8 bit



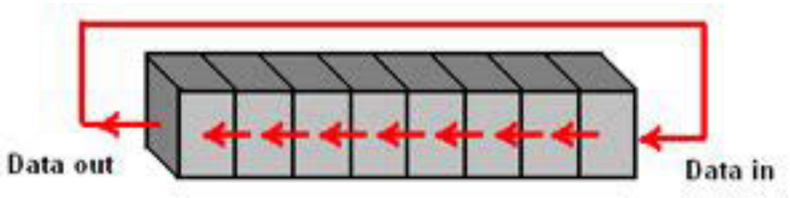
- Operasi paralel input serial out register 8 bit



- Operasi parallel input serial output register 8 bit



- Operasi geser ke kanan register 8 bit



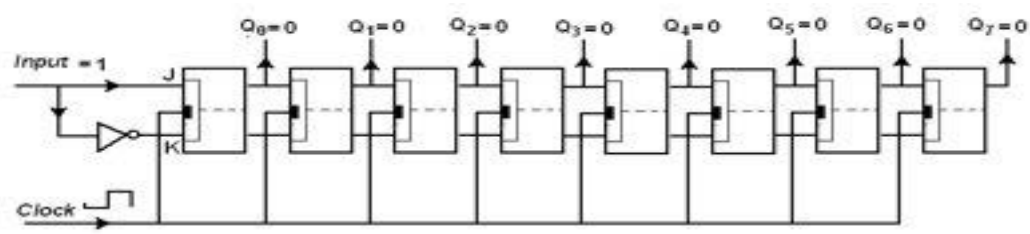
- Operasi geser ke kiri register 8 bit

Gambar 2.24. Operasi register dalam sistem mikroprosesor

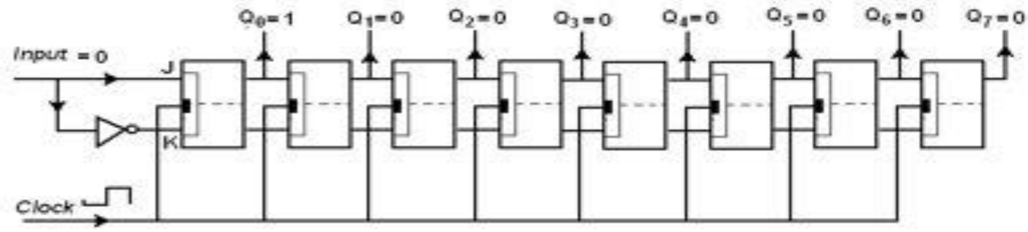
- **Register Masuk Serial Keluar Paralel (Serial In Paralel Out / SIPO).**

Register yang difungsikan sebagai operasi konversi data serial menjadi data parallel disebut dengan istilah SIPO yaitu Serial Input Paralel Output, adapun operasi konversi data serial menjadi data paralel dilakukan dengan memasukan data melalui masukan X, dan keluaran data paralel dibaca dari Q_0 sampai Q_7.

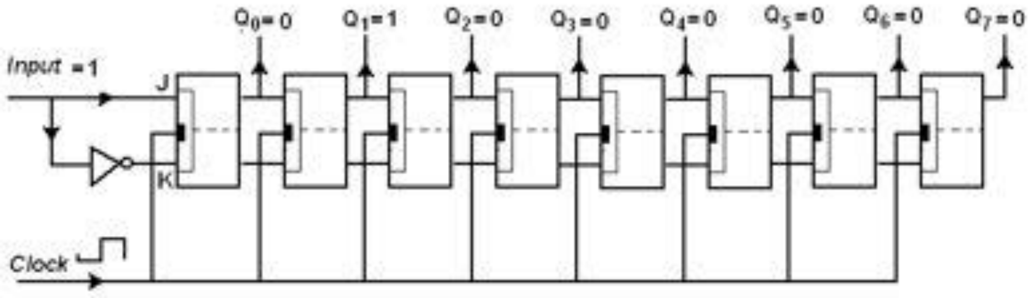
Sebagai ilustrasi proses penyimpanan data 8 bit yang dimasukan melalui saluran input serial dan dikeluarkan secara paralel adalah dengan memberikan clock sebanyak 8 kali sehingga pada akhir clock yang ke delapan bit data tersebut sudah tersimpan pada register secara paralel.



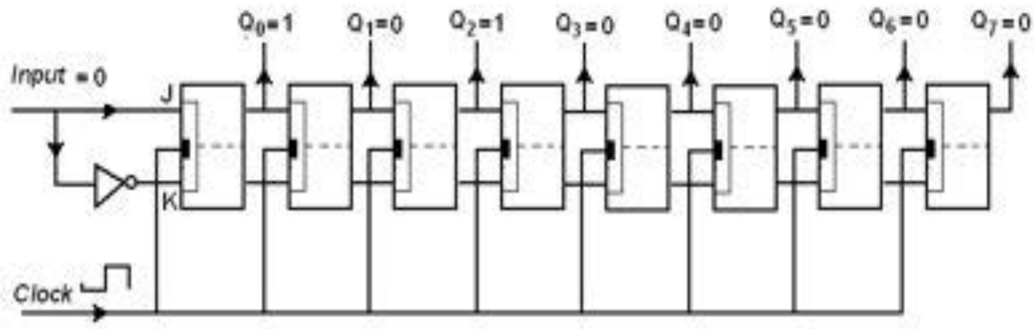
a. Kondisi awal Keluaran register Q_0 sampai Q_7 berlogika 0, input =1



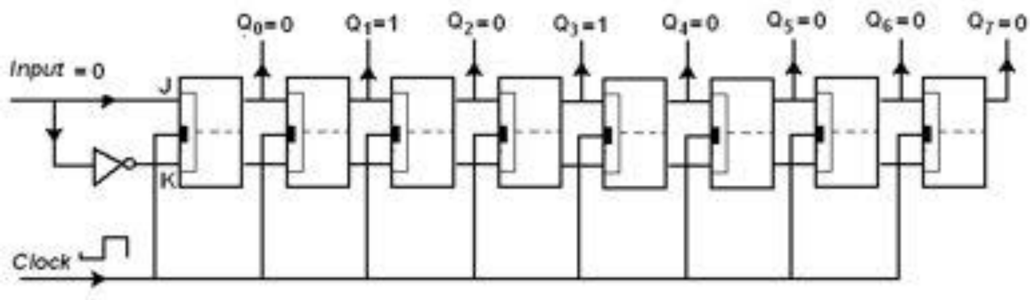
b. Setelah clock pertama Q_0 = 1 dan Q yang lain tetap 0, input diberi 0



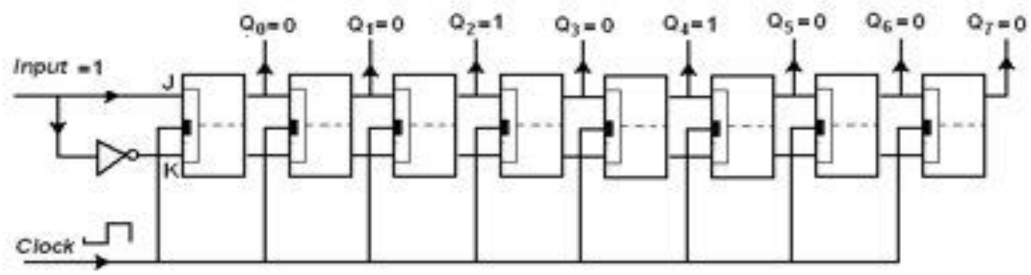
c. Setelah clock kedua Q_0=0, Q_1=1 yang lain tetap 0, input diberi 1



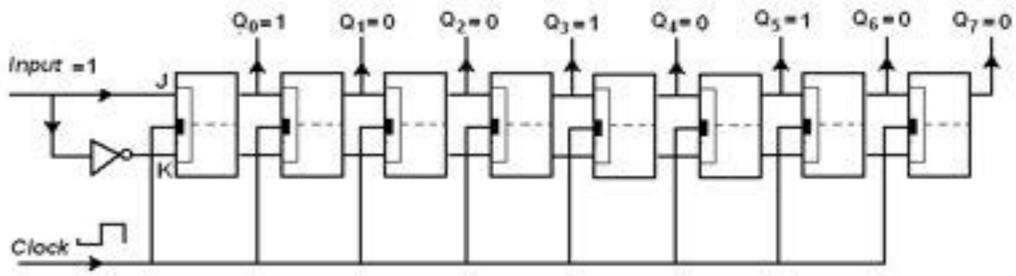
d. Setelah clock ketiga Q_0 =1, Q_1=0 dan Q_2=1 yang lain tetap 0, input diberi 0



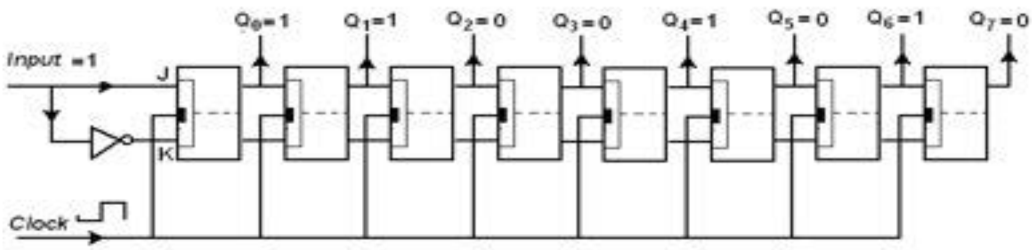
e. Setelah clock keempat Q_0=0, Q_1 =1, Q_2=0, Q_3=1, Q_4..Q_7 =0, input diberi 0.



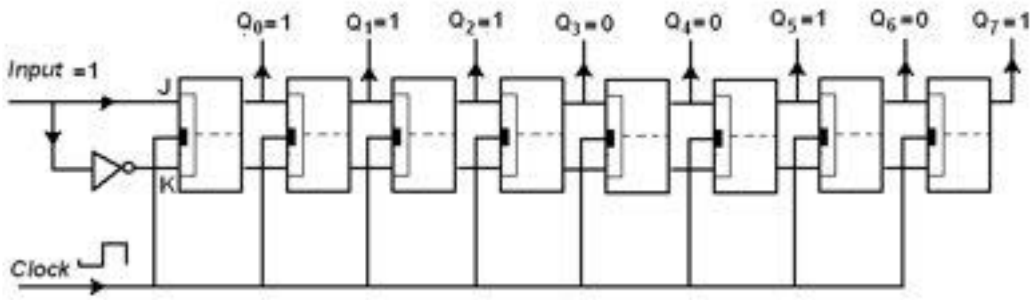
f. Setelah clock kelima Q_0=0, Q_1 =0, Q_2=1, Q_3=0, Q_4=1, Q_5..Q_7 =0, input=1



g. Setelah clcok keenam Q_0=1, Q_1 =0, Q_2=0, Q_3=1, Q_4=0, Q_5=1, Q_6..Q_7 =0, input=1



h. Setelah clcok ketujuh Q_0=1, Q_1 =1, Q_2=0, Q_3=0, Q_4=1, Q_5=0, Q_6=1,Q_7 =0, input=1



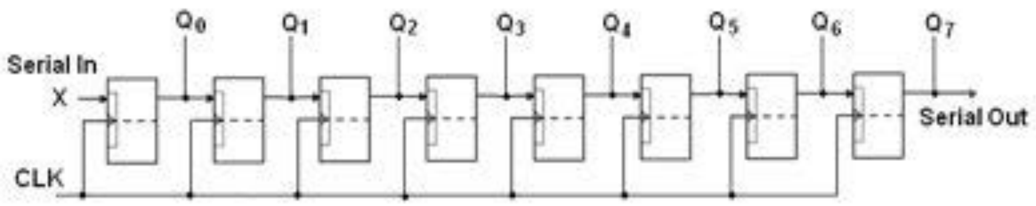
i. Setelah clock kedelapan Q_0=1, Q_1 =1, Q_2=1, Q_3=0, Q_4=0, Q_5=1, Q_6=0,Q_7=1

Gambar 2.25. Proses pemasukan data 8 bit ke dalam penyimpan data 8 bit.

Berdasarkan gambar 2.25 data yang dimasukan secara serial adalah 11100101, dengan memberikan 8 clock maka terbaca secara paralel pada keluaran Q adalah Q_0=1, Q_1 =1, Q_2=1, Q_3=0, Q_4=0, Q_5=1, Q_6=0,Q_7=1. Perubahan pada keluaran Q0 sampai dengan Q7 dilakukan secara bertahap melalui pemberian clock, setiap kali terjadi clock akan terjadi perubahan data pada keluaran. Proses perubahan terjadi melalui pergeseran bit per bit dari data yang dimasukan secara serial, sehingga register ini disebut sebagai register geser.

- **Register Masuk Serial Keluar Serial (Serial In Serial Out / SISO).**

Register yang difungsikan sebagai operasi konversi data serial menjadi data parallel kemudian dikeluarkan lagi secara serial disebut dengan istilah SISO yaitu Serial Input Serial Output, adapun operasi data amsuk secara serial dan data dikeluarkan lagi secara serial dilakukan dengan memasukan data melalui masukan X, dan keluaran data dibaca melalui Q_7 secara serial.



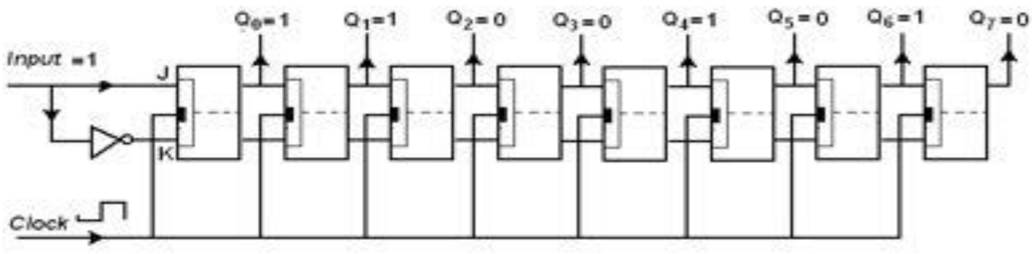
Gambar 2.26. Register Serial In Serial Out

Pada prinsipnya antara register SIPO dengan register SISO memiliki arsitektur yang sama, akan tetapi keluaran register digunakan keluaran paling akhir yaitu Q_7 sehingga keluaran adalah data secara serial. Register dengan arsitektur SIPO dan SISO ini disebut juga dengan konverter data serial ke paralel dan konverter paralel ke serial.

Proses penyimpanan data dilakukan secara 8 bit yang dimasukan melalui saluran input serial dan dikeluarkan secara serial melalui Q7. Pemasukan dan pengeluaran data terjadi dengan memberikan clock, setiap kali clock terjadi pemasukan data 1 bit serial dan bersamaan itu pula terjadi pengeluaran 1 bit. Sehingga untuk data 1 byte diperlukan clock sebanyak 8 kali, sehingga pada

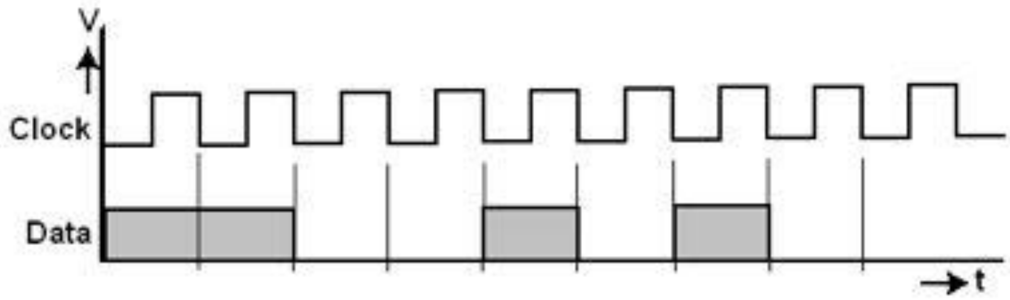


akhir clock yang ke delapan bit data tersebut sudah tersimpan pada register secara paralel dan pada Q_7 data juga sudah dikeluarkan sebanyak 8 bit..



Gambar 2.27. Kondisi awal register Q_0 sampai Q_7 berlogika 11001010

Berdasarkan gambar 2.27 tersimpan data pada register 11001010, jika dikeluarkan paralel dapat diakses secara langsung 8 bit pada Q_0 sampai Q_7. Sedangkan operasi SISO data tersebut dapat diakses pada Q_7, sehingga pada akhir clock kedelapan data yang dikeluarakan adalah 11001010. Berikut bentuk gambar pulsa keluaran serial tersebut:

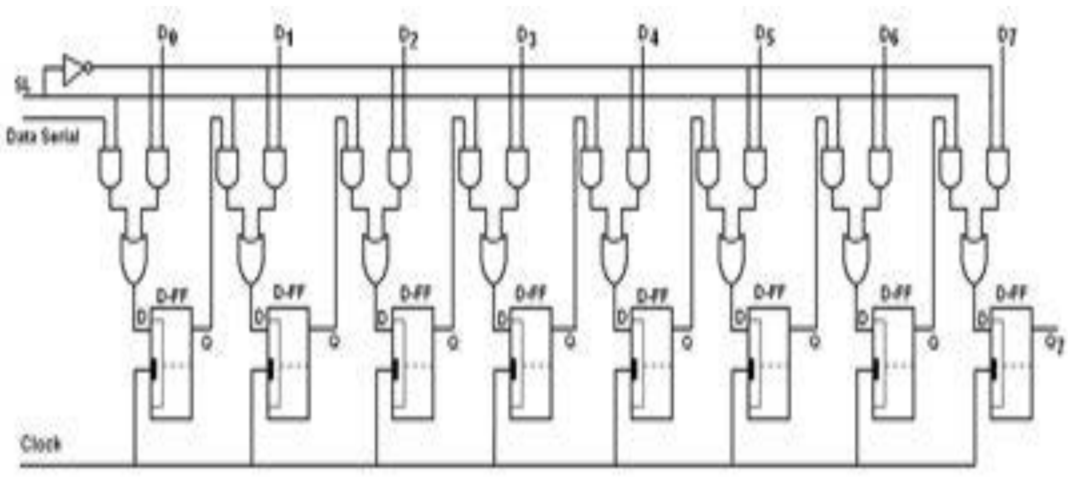


Gambar 2.28. Pulsa data keluaran serial Q_7 pada regiter operasi SISO.

Pada gambar 2.28. data yang semula berada pada register dan tersimpan secara paralel melalui clock dikonversi menjadi data serial, operasi ini dalam mikroprosesor dikenal dengan menggeser bit pada register ke arah kanan.

- **Register Masuk Paralel Keluar Serial (*Parallel In Serial Out* / PISO).**

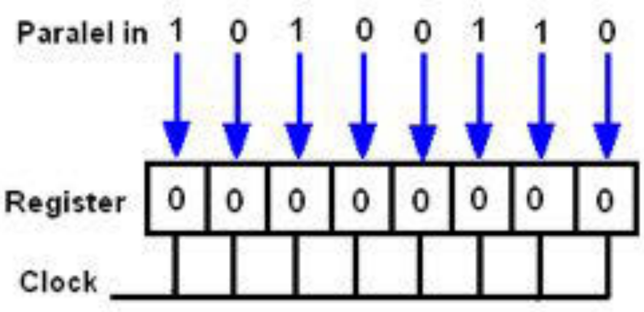
Register yang difungsikan sebagai operasi register dengan masukan parallel dan keluaran serial rangkaian dasarnya dapat dilihat pada gambar 2.29, data secara parallel dapat dimasukan melalui D_1, D_2, D_3 dan D_4 dan dikeluarkan secara serial melalui Q_4.Untuk mengeluarkan data secara serial harus diberikan clock 8 kali agar kedelapan bit yang dimasukan secara parallel dapat dikeluarkan keseluruhannya secara serial.



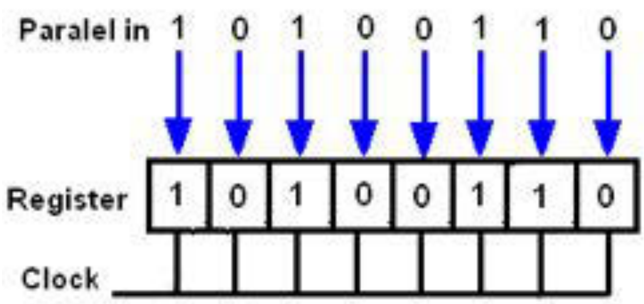
Gambar 2.29. Register Masuk Paralel Keluar Serial (PISO)

Untuk memasukan data secara parallel dikendalikan oleh SL bit, jadi pada saat SL diberi logika 0 dengan disertai satu kali clock maka data secara parallel dimasukan mulai dari D_0 sampai dengan D_7. Sedangkan untuk mengeluarkan bit data yang tersimpan dalam register tersebut harus diberikan logika 1 pada saluran SL, dan data secara serial digeser ke kanan melalui pemberian clock sebanyak 8 kali serta data keluaran dapat dibaca dari keluaran Q_7.

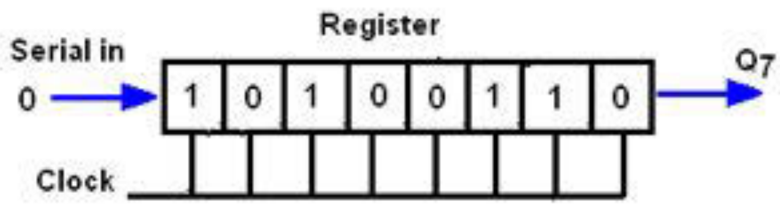
Berikut sebuah contoh ilustrasi proses penyimpanan data 8 bit yang dimasukan melalui saluran input paralel dan dikeluarkan secara serial.



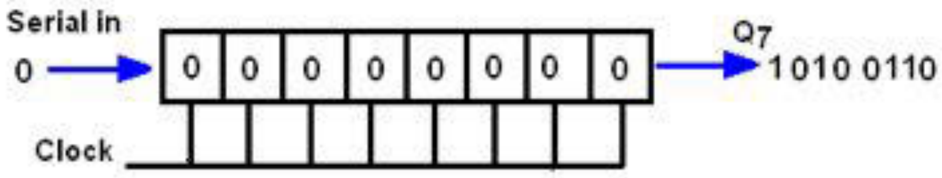
a. Kondisi awal pemasukan data parallel sebelum clock



b. Kondisi pemasukan data parallel setelah clock pertama



c. Kondisi awal sebelum bit digeser ke kanan



d. Kondisi akhir bit digeser 8 kali clock (serial in selalu diberi 0)

Gambar 2.30. Register parallel in serial out

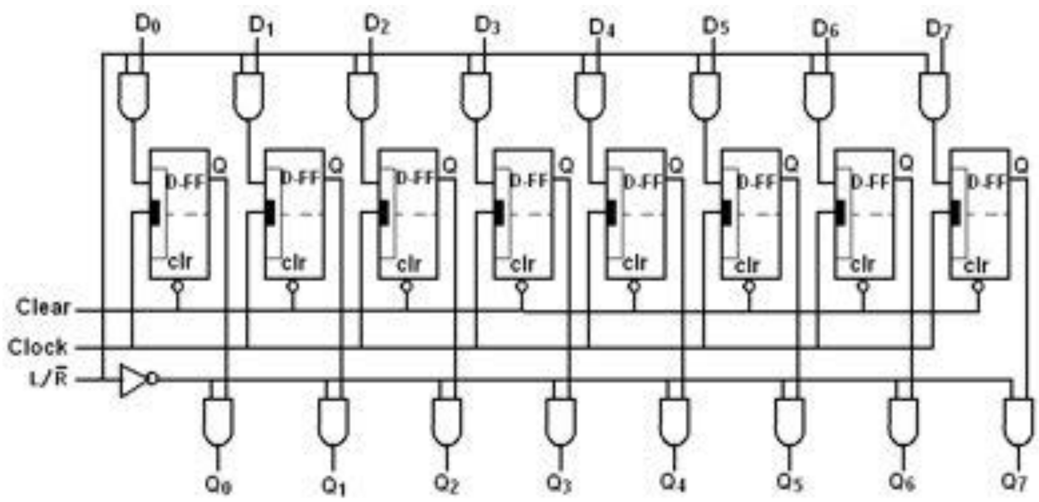
Berdasarkan gambar 2.30, bit data dimasukan melalui saluran parallel in dengan memberikan satu kali clock, dengan demikian maka 8 bit data secara parallel sudah tersimpan dalam register, Sedangkan untuk mengeluarkan data parallel tersebut secara serial adalah dengan menutup saluran input parallel (SL=1), bersamaan dengan itu diberikan data secara serial, dalam contoh data serial input diberi logika 0 selama 8 kali clock.

Dengan memberikan clock sebanyak 8 kali data yang sudah dimasukan secara parallel digeser kea rah kanan sehingga pada akhir clock yang ke delapan bit data tersebut sudah dikeluarkan register secara serial melalui Q_7.

- **Register Masuk Paralel Keluar Paralel (*Parallel In Serial Out* / PIPO).**

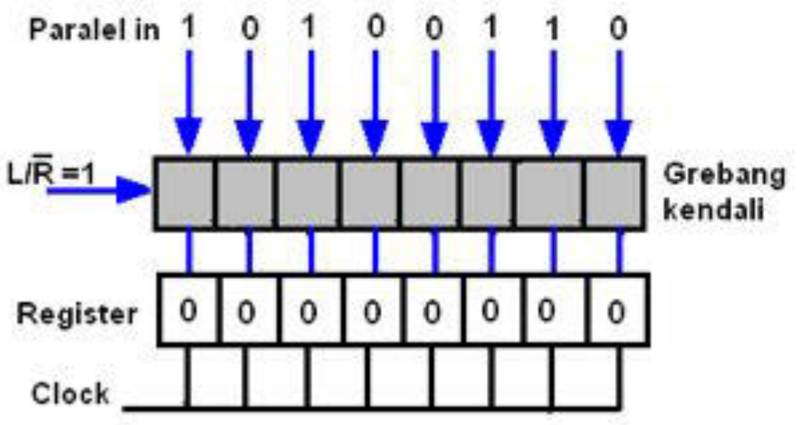
Untuk register parallel in - parallel out (PIPO), semua bit data terbaca secara parallel pada keluaran setelah data diberikan pada masukan secara parallel. Pemasukan bit data secara parallel tersebut dilakukan secara bersamaan, yaitu dengan memberikan satu kali clock maka data akan disimpan oleh flip-flop. Oleh karena bit data jumlahnya adalah 8 bit maka kombinasi nilai yang tersimpan adalah berbatas terendah $(0000\ 0000)_B = (0)_D$ dan batas tertinggi adalah $(1111\ 1111)_B = (255)_D$.

Gambar 2.31 merupakan rangkaian register dengan masukan parallel dan keluaran parallel, sedangkan clock untuk memasukan data diberikan secara bersamaan. Untuk memasukan data secara parallel c$\overline{R}$endalikan oleh L / bit, (L=*load*, R=*read*) jadi pada saat diberi logika 1 dengan disertai satu kali clock maka data secara parallel dimasukan mulai dari D_0 sampai dengan D_7. Sedangkan untuk mengeluarkan bit data yang tersimpan dalam register $\overline{R}$ tersebut harus diberikan logika 0 pada saluran L / , dan data secara parallel dapat dikeluarkan secara serentak (bersamaan) melalui Q_0 sampai Q_7 tanpa pemberian clock. Hal ini bisa terjadi karena untuk setiap keluaran flip=flop di sambung ke gerbang AND, sedar$\overline{R}$jkan inputa gerbang AND satunya lagi disambungkan dengan saluran L / = 0, dengan adanya gerbang NOT maka semua gerbang AND mendapat logika 1 sehingga Q_0 sampai Q_7 sama dengan isi register.

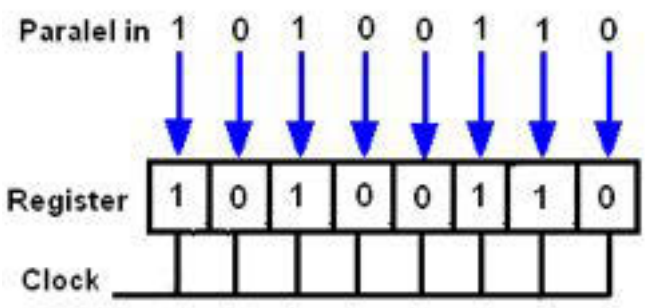


Gambar 2.31. Register *Parallel In-Parallel Out* (PIPO)

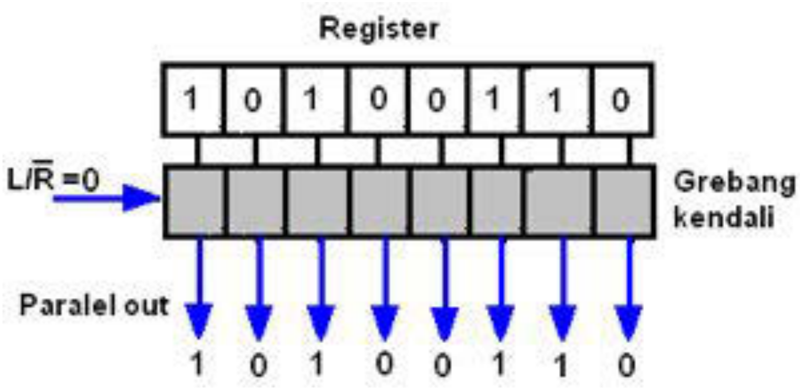
Gambar 2.31 berikut adalah sebuah contoh ilustrasi proses penyimpanan data 8 bit yang dimasukan melalui saluran input paralel dan dikeluarkan secara paralel.



a. Kondisi awal pemasukan data parallel sebelum clock



b. Kondisi pemasukan data parallel setelah clock pertama



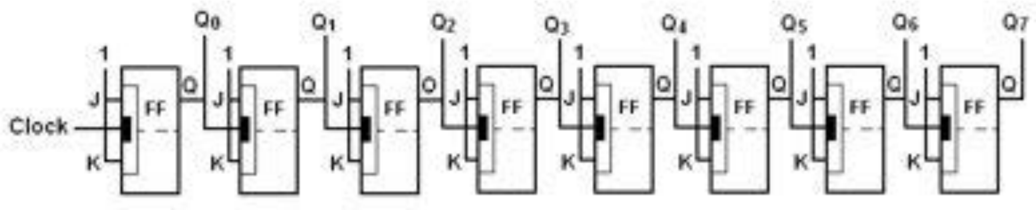
c. Kondisi saat gerbang kendali terbuka (*read*)

Gambar 2.32. Register *parallel in parallel out* (PIPO)

e. Penghitung Biner (Counter)

Salah satu cirikhas kerja JK-FF adalah dengan memberikan logika satu pada masukan J dan masukan K, pada kondisi logika tersebut pada kedua masukan akan memfungsikan JK-FF menjadi sebuah T-FF (*toggle*) sehingga setiap kali clock akan mengubah kondisi keluaran selalu kebalikan dari kondisi logika sebelumnya.

Berbasis pada fungsi tersebut dapat dirangkai sebuah penghitung 8 bit, yaitu semua masukan setiap FF diberi logika 1 (J=1 dan K=1), clock diberikan pada JK-FF pertama sedangkan clock pada JK-FF berikutnya disambungkan ke keluaran JK-FF sebelumnya (lihat gambar 2.33). Setiap kali clock akan membuat nilai keluaran penghitung bertambah satu, jika awalnya 0 maka hitungan berikutnya adalah satu, kemudian dua, tiga dan bertambah terus dengan satu setiap kali selesai clock sampai mencapai 255 kemudian kembali lagi dari 0.



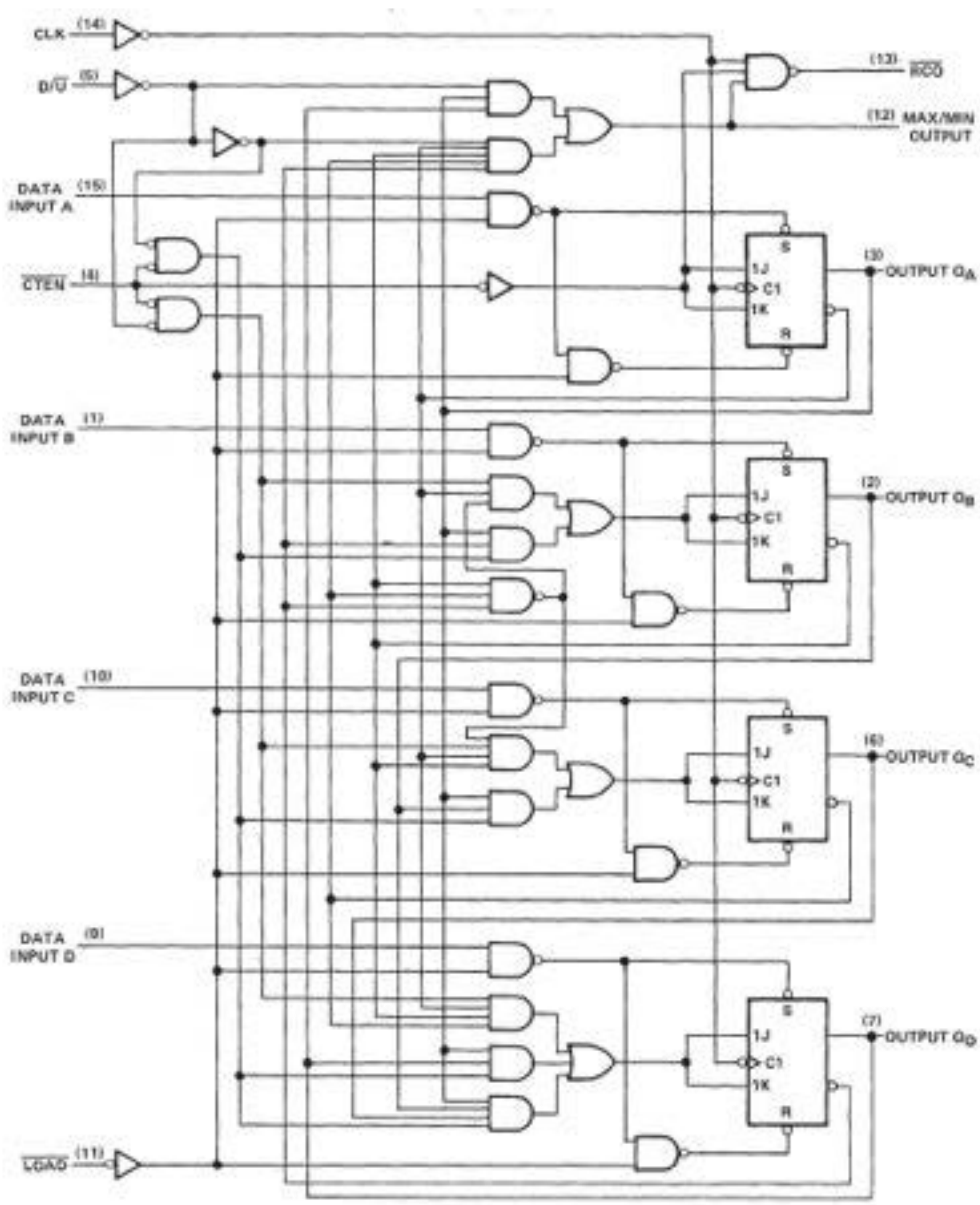
Gambar 2.33. Rangkaian penghitung naik 8 bit.

Misal kondisi awal data yang tersimpan pada penghitung adalah 0 artinya semua FF dalam kondisi reset, dan dengan masukan JK-FF dihubungkan ke logika satu maka semua JK-FF berfungsi sebagai T-FF. Dengan demikian clock pertama akan membuat Q_0 berubah dari 0 ke 1, sedangkan FF yang lain belum mendapat clock dan hanya FF kedua yang baru menerima ½ clock sehingga Q_1 juga belum berubah dan data penghitung adalah $(0000\ 0001)_B$,

Clock kedua membuat toggle pada Q_0 dari 1 menjadi 0, hal ini berakibat pada Q_1 berubah dari 0 menjadi 1 dan FF yang lain masih tetap dan data penghitung adalah $(0000\ 0010)_B$.



Clock ketiga membuat toggle pada Q0 dari 0 menjadi 1, kembali FF kedua baru menerima ½ clock sedang FF yang lain masih tetap sehingga Q_1 sampai Q_7 mempunyai nilai tetap dan data penghitung menjadi $(0000\ 0011)_B$.

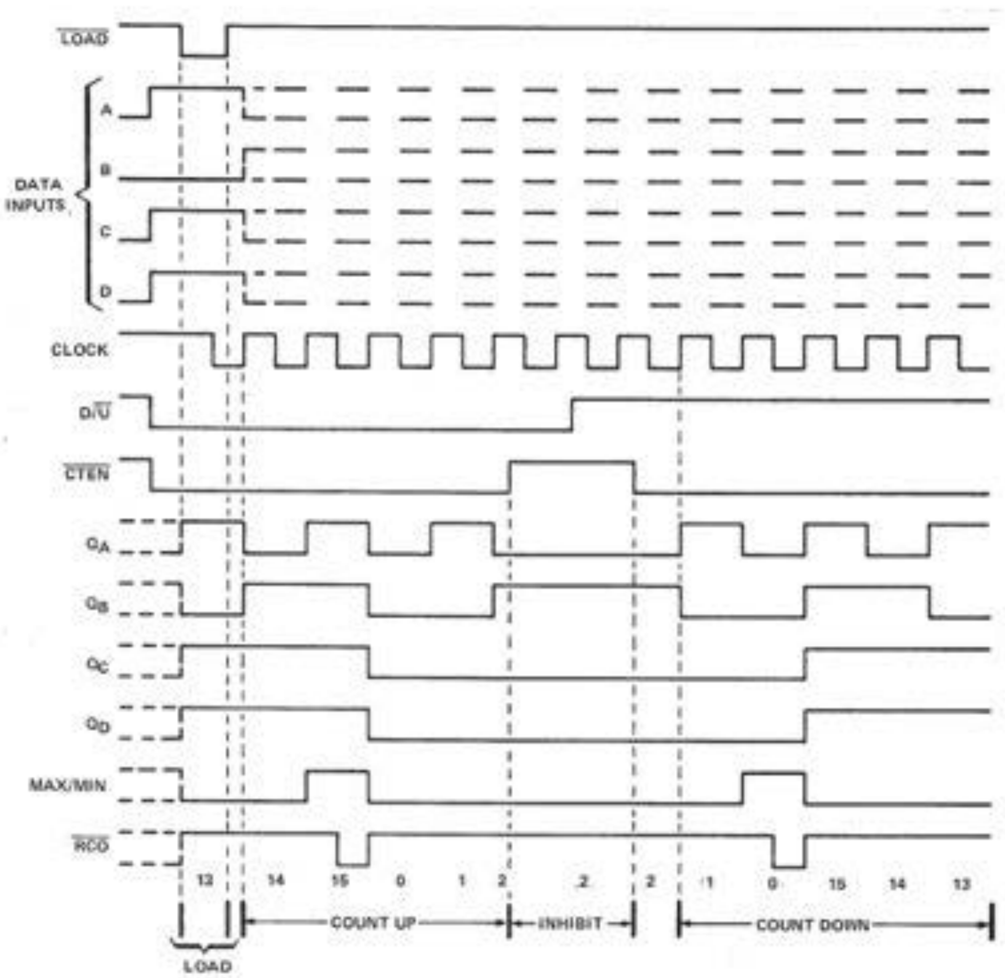


Gambar 2.34. Rangkaian pengitung naik/turun 4 bit (Texas Instrumen)

Gambar 2.34. merupakan contoh sebuah penghitung 4(empat) bit naik/turun yang sudah dikemas dalam bentuk IC dengan tipe 74191, flip-flop yang digunakan merupakan master-slave FF. Penghitung dapat diprogram

sehingga data keluaran dapat disesuaikan dengan data masukan, data masukan digunakan untuk men-set flip-flop dan untuk membuat keluaran sama dengan 0 dilakukan reset pada saluran reset dari FF.

Rangkaian ini dibuat untuk dapat melakukan hitungan naik atau hitungan turun, dan prinsip kerja inilah yang digunakan oleh mikroprosesor untuk melakukan hitungan naik satu (inkrimen) dan hitungan mundur satu (dikrimen).



Gambar 2.35. Diagram Pulsa Penghitung naik/turun (up/down counter).

(Texas Instrumen, Datasheet 7490)

Gambar 2.35 memberikan ilustrasi sekuensial proses load (memuati penghitung), menghitung dan inhibit. Adapun sekuensial proses hitung dapat dilakukan dengan urutan sebagai berikut:

- Preset, yaitu memuati data pada setiap flip-flop pembentuk penghitung, yaitu dengan memberikan logika 0 pada saluran Load.
- Penghitung naik (count up) dapat melakukan hitungan sampai delapan dan sembilan maksimum, kemudian kembali ke nol, satu, dua dan seterusnya.
- Inhibit yaitu mempertahankan nilai hasil penghitungan dan untuk melakukannya dengan cara memberi logika nol pada saluran CTEN.
- Penghitung turun (Count down), yaitu dengan memberikan logika 1 pada saluran D/U, hitungan satu, nol, kemudian kembali ke Sembilan, delapan, tujuh dan seterusnya.

Untuk bisa menghitung diperlukan sebuah clock, jadi melalui clock inilah penghitung melakukan penghitungan.

2.2. RANGKAIAN PEMBENTUK UNIT ARITMATIK LOGIK (ALU)

Mode operasi suatu komputer pada dasarnya dapat diketahui dengan melihat kemampuannya dalam memproses data yang diinstruksikan oleh suatu program. Artinya dengan program yang berbeda suatu komputer dapat digunakan untuk menyelesaikan masalah yang berbeda.

Beberapa tahun yang lalu komputer hanya dipergunakan untuk instalasi yang sangat terbatas, sekarang dengan teknologi semikonduktor komputer dapat dibuat menjadi lebih kecil dan lebih murah sehingga penggunaannya menjadi sangat luas tak terbatas. Kompter kecil ini disaebut komputer mini.

Langkah berikutnya dalam pengembangan teknologi pembuatan komputer adalah penggunaan komponen LSI (Large Scale Integration) yang diterapkan pada CPU (Central Processing Unit) suatu mikrokonmputer yang didalamnya terdapat ribuan komponen semikonduktor hanya dalam satu chip CPU. CPU suatu mikrokomputer disebut mikroprocesor. Salah satu jenis mikroprosesor

yang dipergunakan dalam eksperimen ini adalah Intel 8080 didalamnya terdapat lebih dari 4500 transistor MOS dalam sattu chip yang berukuran 23 mm^2. Dengan adanya lebih dari 4500 transistor memungkinkan untuk mengimplementasikan suatu kontrol dan aritmetic logic unit yang lengkap untuk mikrokomputer. Dalam aritmetic logic unit, operasi aritmetika dan logika dapat dilakukan karena adanya control unit yang mengontrol prosesor internal dalam komputer. Agar mikroprosesor dapat bekerja disuatu mikrokomputer maka diperlukan perangkat tambahan seperti memory program, memory data, input output port, tambahan rangkaian digital dal lain sebagainya tergantung dari aplikasi individual. Pada prakteknya tidak ada suatu masalah yang dapat diselesaikan oleh satu mikroprosesor saja, perangkat tambahan pasti dibutuhkan. Tetapi bahwa mikroprosesor pasti selalu diperlukan.

Dalam pembahasan berikut ini akan dijelaskan prinsip kerja mode operasi suatu mikrokomputer secara langkah demi langkah.

2.2.1. Adder

Adder adalah rangkaan digital yang memiliki fungsi sebagai penjumlah biner.

A + B	Penjumlah	Carry
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1

Rangkaian digital yang dapat melakukan operasi aritmetika yang hasil keluaran-nya seperti tabel di atas disebut Half adder.

Tabel kebenaran Penjumlah setengah

Variabel Masukan		Hasil Penjumlahan	
B	A	∑ (SUM)	U (Carry)
0	0	0	0

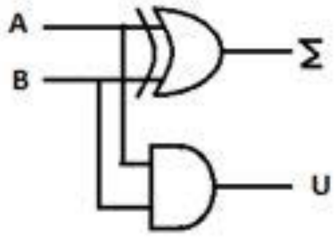
0	1	1	0
1	0	1	0
1	1	0	1



Gambar 2.36. Diagram blok Penjumlah setengah

Dari tabel kebenaran dapat diperoleh persamaan fungsi logika untuk keluaran penjumlah Σ dan carry U sebagai beikut :

Penjumlah	$\Sigma = (A \wedge \overline{B}) \vee (\overline{A} \wedge B) = A \veebar B$	(EXCLUSIVE-OR)
Output Carry	$U = (A \wedge B)$	(AND)



Gambar 2.37. Rangkaian Half Adder

Tabel kebenaran Penjumlah setengah

Variabel Masukan		Hasil Penjumlahan	
B	A	∑ (SUM)	U (Carry)
0	0	0	0
0	1	1	0

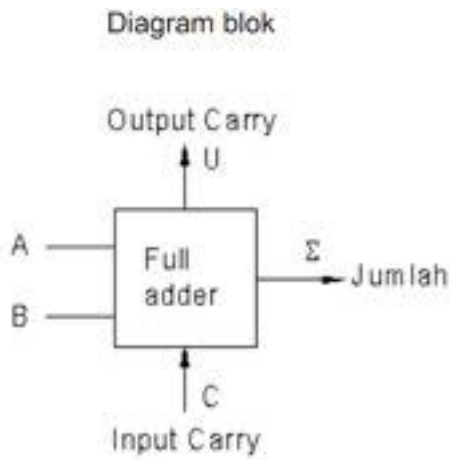
1	0	1	0
1	1	0	1

Dari persamaan di atas dapat ditemukan gambar rangkaian digital seperti diperlihatkan pada Gambar 2, selanjutnya jika bit yang dijumlahkan lebih dari 1 misal 4 bit maka variabel menjadi A_0, A_1, A_2, A_3 dan B_0, B_1, B_2, B_3. Demikian juga rangkaian penghitung merupakan kaskada dari penghitung setengah, untuk penghitung lebih dari 1 bit yang dijumlahkan disebut dengan penghitung penuh.

Jika jumlah bit lebih banyak kombinasi (disebut word), penggunaan half adder tidak dapat dilakukan karena tidak ada masukan carry. Untuk itu kita harus menambahkan masukan carry pada half adder. Rangkaian penjumlah yang dengan tambahan masukan carry ini disebut full adder.

Tabel kebenaran

Masukan			Keluaran	
A	B	C	Σ	U
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Gambar 2.38. Full Adder

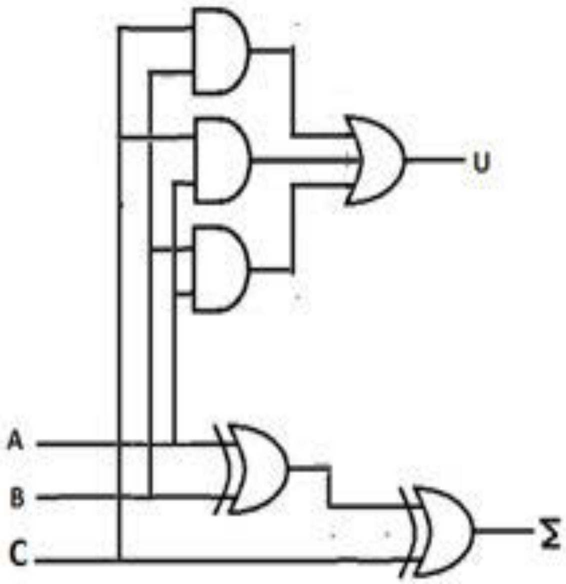
Persamaan fungsi logika untuk keluaran jumlah full adder adalah

Penjumlah $\Sigma = (\bar{A} \wedge B \wedge \bar{C}) \vee (A \wedge \bar{B} \wedge \bar{C}) \vee (\bar{A} \wedge \bar{B} \wedge C) \vee (A \wedge B \wedge C)$

$= (A \veebar B) \veebar A$

Output Carry $U = (A \wedge B \wedge \bar{C}) \vee (\bar{A} \wedge B \wedge C) \vee (A \wedge \bar{B} \wedge C) \vee (A \wedge B \wedge C)$

$= (A \wedge B) \vee (B \wedge C) \vee (A \wedge C)$



Gambar 2.39. Rangkaian Full Adder

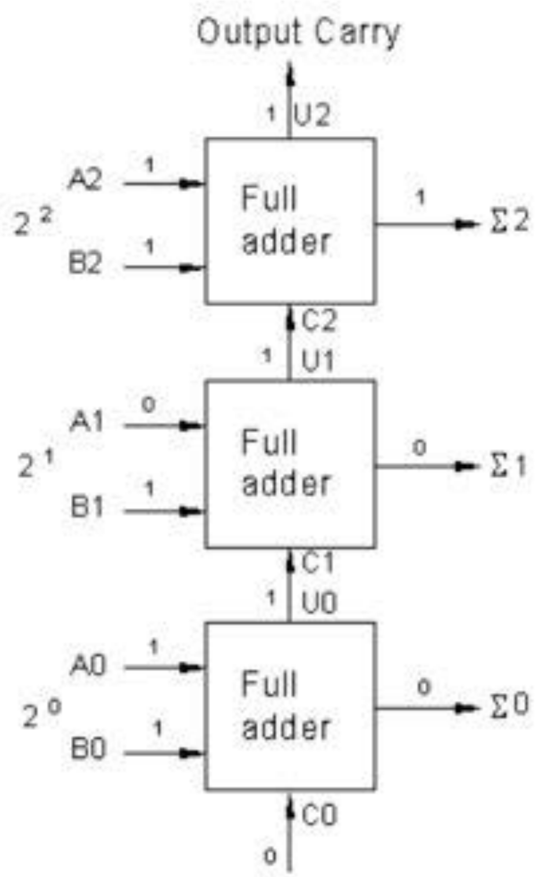
Jika dua n-bit word dijumlahkan diperlukan beberapa full adder yang saling disambungkan, sebagai contoh sederhana penjumlahan 3 bit word diperlihat-kan di bawah ini.

		2^2	2^1	2^0
A		1	0	1
B		1	1	1
U	1	1	1	
Σ		1	0	0

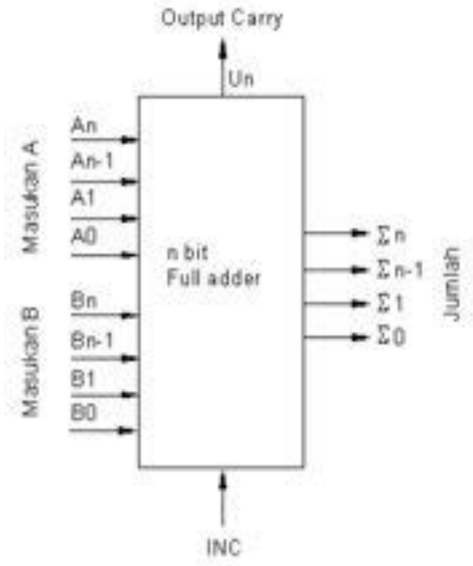
Pada prakteknya penjumlahan 3 bit word memerlukan tiga buah full adder yang disambungkan secara berurutan (sekuensial), masukan carry C0 diset =

0 artinya pada penjumlah level bawah tidak ada carry seperti tampak pada gambar 2.40.

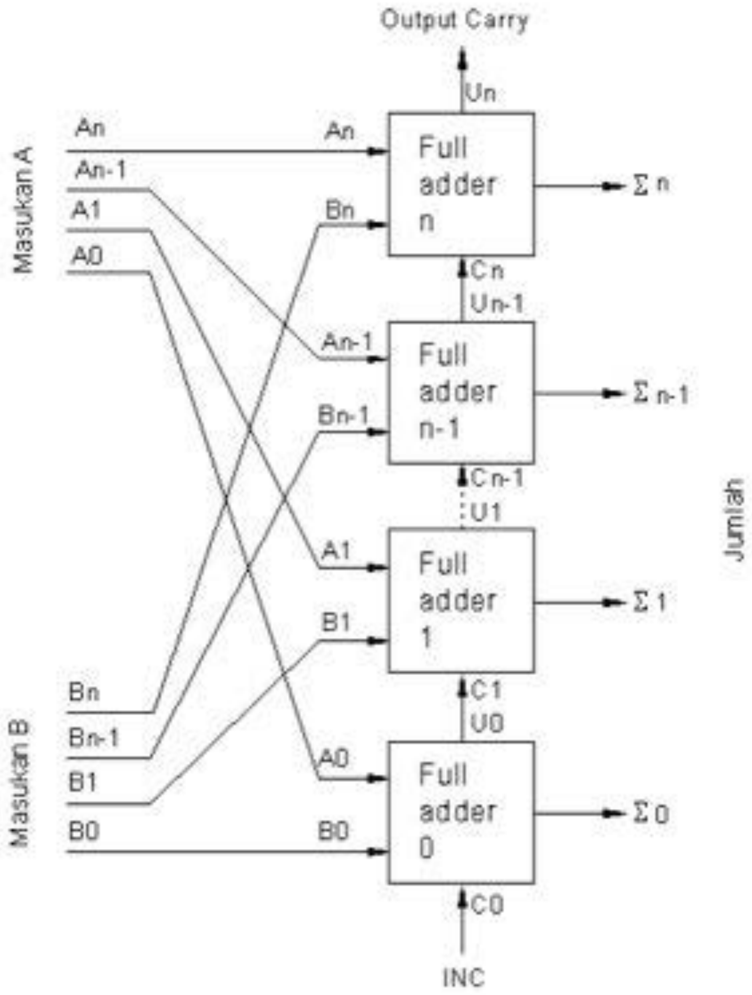
Apabila masukan C0 kita set = 1 maka hasil penjumlahan akan bertambah 1. Dalam praktek penambahan satu ini sangat penting untuk bererapa aplikasi. Masukan carry C0 digunakan untuk masukan increment (INC) dan suatu rangkaian penjumlah digital yang dilengkapi dengan sebuah masukan incremen disebut ripple-carry adder.



Gambar 2.40. Penjumlah biner 3 bit



Gambar 2.41. Diagram blok ripple-carry adder



Gambar 2.42. Rangkaian digital ripple-carry adder

2.2.2. Pengurang (*Subtractor*)

Bilangan 8-bit, atau isi register, atau isi lokasi memori tertentu dapat dikurangkan dengan isi dari akumulator dan hasil dari proses pengurangan tersebut disimpan ke dalam akumulator. Prinsip pengurangan bilangan biner adalah dengan penerapan komplemen-2, demikian juga pada ALU untuk pengurangan dua bilangan juga diterapkan komplemen-2 sehingga terbentuk bilangan negatip. Dalam melaksanakan pengurangan data antara dua register tidak dapat dilakukan secara langsung.



Hukum dasar Komplemen-1 yaitu angka nol diubah menjadi satu dan satu menjadi nol.

Contoh komplemen -1 bilangan 15 dalam format data 8 bit

+15 = 00001111

-15 = 11110000

Hukum dasar Komplemen-2 yaitu komplemen-1 ditambah 1

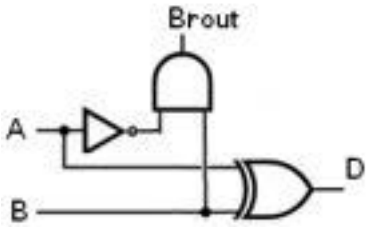
Contoh komplemen-2 bilangan 15 dalam format data 8 bit

-15 = 11111000 (Komplemen-1)

1 +

11111001 (Komplemen-2)

Rangkaian untuk pengurang setengah dapat digambarkan sebagai berikut:



Gambar 7. Pengurang Setengah (*Half Subtractor*)

Tabel Kebenaran

B	A	D	Br
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

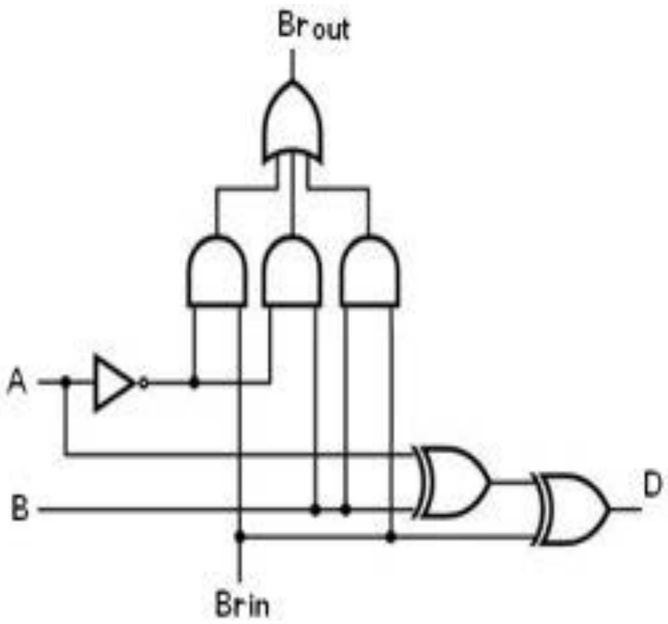
Berdasarkan tabel kebenaran persamaan pengurang setengah dapat dituliskan sebagai berikut:

$D = A \oplus B$

$Br = \overline{B} . A$

dimana: D = beda, A dan B = variabel, Br = Pinjaman (Borow)

Rangkaian untuk pengurang penuh dapat digambarkan sebagai berikut:



Gambar 8. Pengurang Penuh (*Full Subtractor*)

Tabel Kebenaran

A	B	Brin	D	Brout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Berdasarkan tabel kebenaran persamaan pengurang penuh dapat ditulis-kan sebagai berikut:



$$D = (A \oplus B) \oplus Brin$$

$$Br = C.\overline{(A \oplus B)} + \bar{A}.B$$

dimana: D = beda, A dan B = variabel, Br = Pinjaman (Borow)

Tabel Kebenaran

A	B	Brin	D	Brout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

2.2.3. Adder/Subtracter

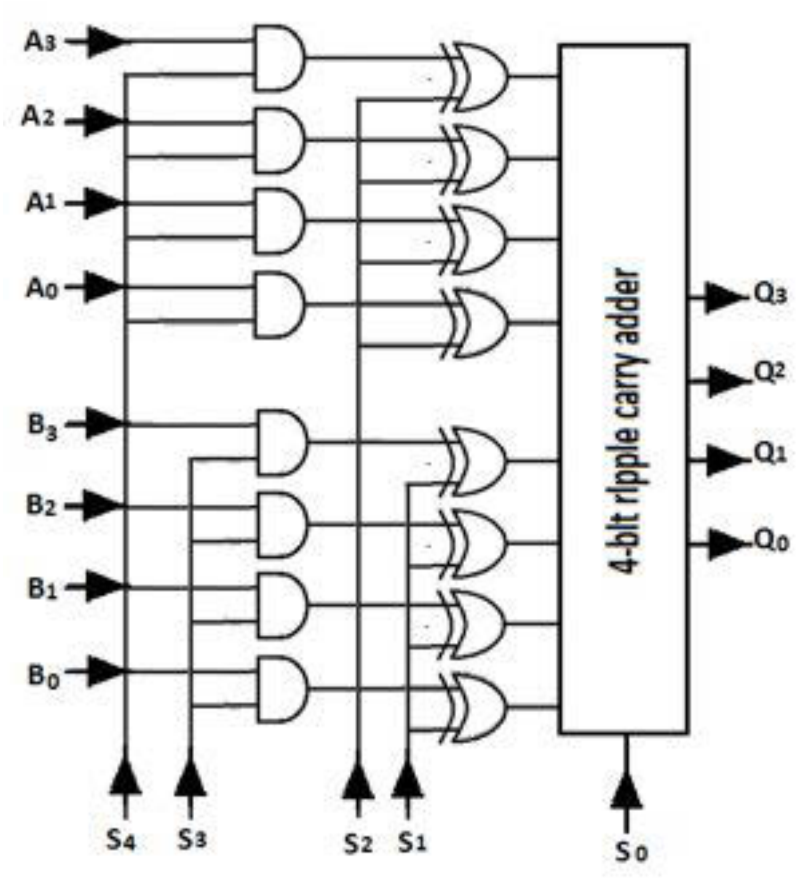
Rangkaian penjumlah Gambar 6. dapat dikembangkan dengan menambahkan gerbang AND dan EXOR pada masukannya sehingga beberapa fungsi rangkaian berkembang tidak hanya sebagai penjumlah melainkan berfungsi pula sebagai rangkaian pengurang, sehingga rangkaian ini disebut adder/substractor.

Rangkaian adder/subtractor berikut memiliki 5 masukan kontrol S4 sampai S0 yang dipergunakan untuk memilih operasi. Tabel Fungsi adder dan subtractor. memperlihatkan variasi masukan kontrol S4 sampai S0 dan fungsi keluaran.

Dari tabel fungsi adder dan subtractor terdapat 32 kemungkinan fungsi yang dapat dioperasikan.

Kita ambil salah satu contoh dari kemungkinan yang diperlihtakan pada tabel diatas misalnya fungsi kontrol S4 sampai dengan S0 = 1 1 0 0 0 menghasilkan

fungsi A + B. Jika S3 dan S4 diset pada 0 maka masukan A dan B akan mati karena setiap gerbang AND tersambung pada setiap masukan A dan B sedangkan amsukan gerbang lainnya tersambung ke saklar masukan kontrol S3 untuk masukan A dan S2 untuk masukan B, sehingga ketika S3 = 0 maka apapun masukan A akan diset = 0.



Gambar 2.43. Adder dan subtractor 4 bit

Dari tabel fungsi adder dan subtractor terdapat 32 kemungkinan fungsi yang dapat dioperasikan.

Kita ambil salah satu contoh dari kemungkinan yang diperlihtakan pada tabel diatas misalnya fungsi kontrol S4 sampai dengan S0 = 1 1 0 0 0 menghasilkan fungsi A + B. Jika S3 dan S4 diset pada 0 maka masukan A dan B akan mati karena setiap gerbang AND tersambung pada setiap masukan A dan B



sedangkan amsukan gerbang lainnya tersambung ke saklar masukan kontrol S3 untuk masukan A dan S2 untuk masukan B, sehingga ketika S3 = 0 maka apapun masukan A akan diset = 0.

Ketika S2 = 0 maka masukan B akan diset = 0. Dengan demikian fungsi kontrol S3 dan S2 adalah untuk meloloskan data masukan ke tahap berikutnya untuk diproses atau tidak oleh blok penjumlah. Setelah keluar dari gerbang AND sebagai pelolos data, setiap keluaran AND tersambung pada gerbang XOR yang mana setiap gerbang XOR tersebut salah satu masukan lainnya tersambung pada saklar masukan kontrol S2 dan S1. S2 dipergunakan untukmengontrol masukan A dan S1 dipergunakan untuk kendali masukan B.

Tabel Fungsi adder dan subtractor

S4	S3	S2	S1	S0	Fungsi keluaran
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	-1
0	0	0	1	1	0
0	0	1	0	0	-1
0	0	1	0	1	0
0	0	1	1	0	-2
0	0	1	1	1	-1
0	1	0	0	0	B
0	1	0	0	1	B + 1
0	1	0	1	0	$-B-1=\overline{B}$
0	1	0	1	1	- B
0	1	1	0	0	B - 1
0	1	1	0	1	B
0	1	1	1	0	$-B-2=\overline{B}-1$
0	1	1	1	1	$-B-1=\overline{B}$
1	0	0	0	0	A
1	0	0	0	1	A + 1
1	0	0	1	0	A - 1
1	0	0	1	1	A
1	0	1	0	0	$-A-1=\overline{A}$
1	0	1	0	1	- A
1	0	1	1	0	- A - 2
1	0	1	1	1	$-A-1=\overline{A}$
1	1	0	0	0	A + B
1	1	0	0	1	A + B + 1
1	1	0	1	0	A - B - 1
1	1	0	1	1	A - B
1	1	1	0	0	B - A - 1
1	1	1	0	1	B - A
1	1	1	1	0	- A - B -2
1	1	1	1	1	- A - B - 1

Jika S2 = 0 maka keluaran gerbang EXOR adalah A XOR 0 = A dan jika S2 = 1 maka keluaran gerbang EXOR adalah A XOR 1 = A. dengan demikian ketika S2 = 1 masukan A akan dibalik (komplemen satu).

Hal yang sama berlaku juga untuk masukan S1 yang mengontrol masukan B untuk fungsi komplemen satu. Saklar S0 merupakan masukan carry untuk rangkaian penjumlah yang berfungsi sebagai masukan incremen (INC). S0 sangat diperlukan pada

operasi pengurangan untuk mendapatkan komplemen dua. Untuk fungsi kontrol S4 sampai dengan S0 = 0 0 0 1 0 , S1 =1 dan masukan kontrol yang lainnya adalah 0, ini berarti semua gerbang keluaran pada keempat gerbang XOR bagian atas adalah 0. Sementara 4 gerbang XOR menghasilkan 1 dan pada bagian adder, penjumlahan akan dlakukan seperti yang dibawah ini :

A		0 0 0 0
B	+	1 1 1 1
Keluaran		1 1 1 1

Hasil tersebut berarti sama dengan –1 pada aritmatika komplemen dua. Ketika fungsi kontrol

S4 sampai dengan S0 diset = 11011, fungsi keluaran pengurangan A – B.

Penjelasan per saklar kontrol sebagai beikut :

S4 = 1 Data masukan A diloloskan

S3 = 1 Data masukan B diloloskan

S2 = 0 Data masukan A tidak dibalik (tidak di komplemen satu)

S1 = 1 Data masukan B dibalik (komplemen satu)

S0 = 1 Incremen 1 (+1)

Berdasarkan kombinasi kelima saklar yaitu S0 sampai dengan S4 akan membentuk fungsi keluaran sebagai berikut:

$$A + B + 1 = A - B$$

2.2.4. Arithmetic Logic Unit (ALU)

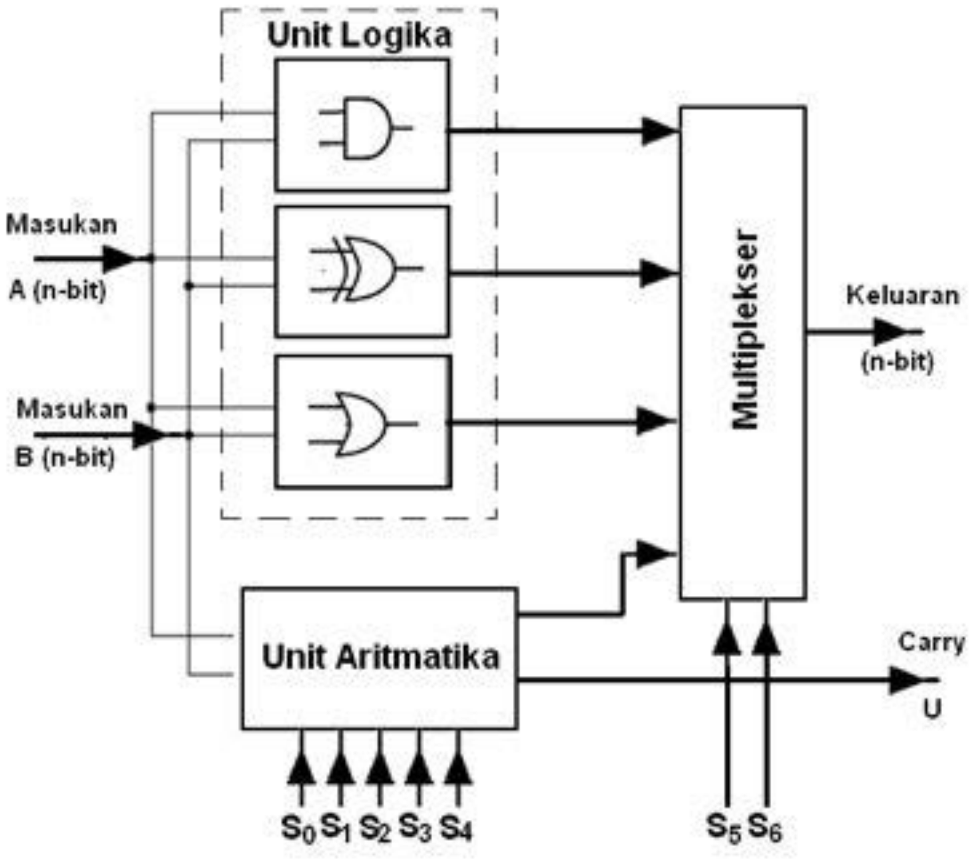
Agar mikroprosesor tidak hanya dapat melakukan operasi aritmatika tetapi juga dapat juga melakukan operasi fungsi logika, maka kita harus

mengembangkan rangkaian adder/subtractor dengan menambahkan gerbang logika EXOR, OR dan AND serta sebuah multiplexser. Dengan adanya tambahan tiga gerbang logika tersebut sekarang operasi logika XOR, OR dan AND dapat dilakukan misalnya

Keluaran = A AND B

Keluaran = A OR B

Keluaran = A XOR B

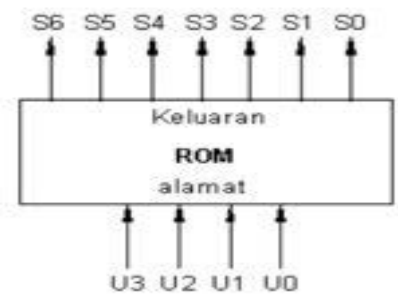


Gambar 2.44. Rangkaian ALU

Masukan kontrol S6 dan S5 adalah kontrol multiplekser yang dipergunakan untuk memilih operasi aritemetika atau logika. Jika S6 = 0 dan S5 = 0 operasi adalah fungsi aritmetika. Pada saat S6 dan S5 pada kondisi yang lain maka

operasi adalah fungsi logika dan selama fungsi logika maka kontrol S4 sampai dengan S0 tidak berpengaruh karena kontrol S4 sampai dengan S0 adalah kontrol untuk operasi aritmetika. Pada prinsipnya dengan kontrol sebanyak 7 bit (S6 sampai dengan S0) seharusnya terdapat 27 = 128 variasi fungsi tetapi tidak semua variasi tersebut diperlukan.

Perhatikan tabel fungsi adder/subtracter terdapat 32 fungsi dan terjadi pengulangan fungsi yang sama dan sebagian besar tidak begitu penting. Untuk itu kita harus membatasi fungsi yang penting saja dengan cara menggunakan ROM. Didalam ROM disimpan data-data kontrol untuk S6 sampai S0 pada alamat alamat tertentu. Pada rancangan ALU ini kita batasi fungsi yang disediakan adalah 13 fungsi dan dikodekan dalam 4 masukan kontrol saja yaitu U3 sampai U0. Sesungguhnya dalam ROM U3 sampai U0 ini adalah jalur alamat sedangkan kode operasi adalah data pada suatu lokasi memory. Contoh untuk instruksi aritmetika A + B kode instruksi dalam table fungsi ALU adalah U3 = 0, U2 = 1, U1 = 1 dan U0 = 0, kalau kita cermati maka kode tersebut adalah alamat pada ROM 01012 sedangkan untuk operasi A + B, masukan kontrol untuk S_0 sampai S_6 adalah 0011000 (S6=S5=0 dan S4 sampai dengan S0 lihat table fungsi adder/subtracter operasi A+B). Dengan demikian kita dapat mengetahui bahwa isi ROM pada alamat 01012 adalah 00110002. Sekarang kita mengenal kode instruksi yang yang disimpan pada ROM dan tidak penting lagi untuk mengetahu kontrol yang harus diberikan kerangkaian yang sebenarnya.

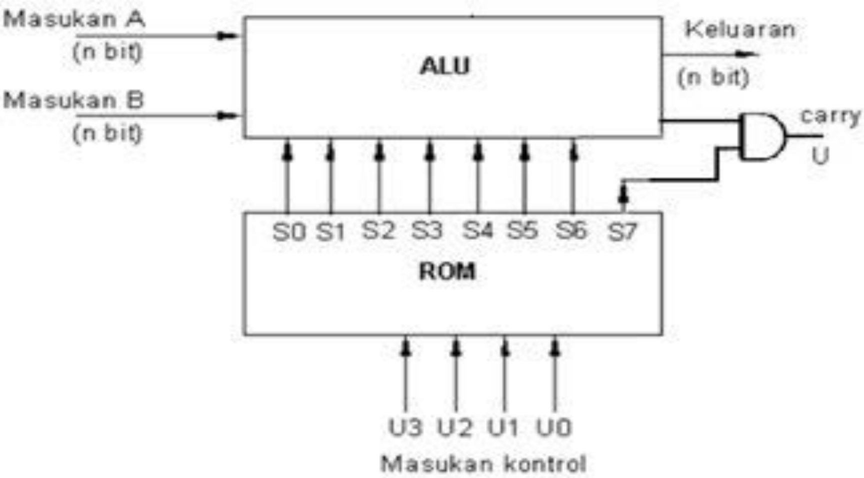


Gambar 2.45. ROM untuk mengkonversi kode instruksi ke masukan kontrol

Table Fungsi ALU

U3	U2	U1	U0	Fungsi keluaran
0	0	0	0	A
0	0	0	1	1
0	0	1	0	$\overline{A}$
0	0	1	1	B
0	1	0	0	0
0	1	0	1	A + 1
0	1	1	0	A - 1
0	1	1	1	A + B
1	0	0	0	A - B
1	0	0	1	A AND B
1	0	1	0	A OR B
1	0	1	1	A XOR B
1	1	0	0	- 1
1	1	0	1	
1	1	1	0	
1	1	1	1	

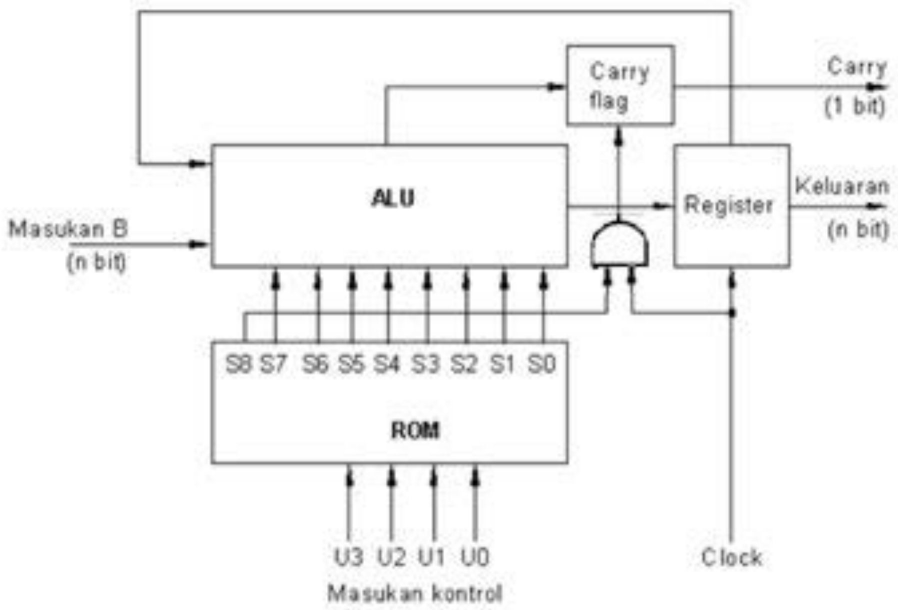
Dengan jumlah saluran kontrol 4 bit terdapat 16 kemungkinan fungsi yang termasuk 3 fungsi untuk pengembangan sistem nantinya. Rangkaian ALU yang dilengkapi dengan konversi kode untuk masukan kontrol ditunjukkan dalam Gambar 2.46. berikut ini.



Gambar 2.46. ALU dengan konversi kode

2.2.5. Akumulator (ACCU)

Akumulator (yang disingkat menjadi accu) adalah tingkatan selanjutnya dari ALU. Berikut adalah gambar accumulator dengan carry flag.



Gambar 2.47. Accumulator dengan carry flag

Terpisah dari fungsi yang telah ditunjukkan pada gambar 11 rangkaian ini terdiri atas register dan carry flag sebagai komponen tambahan yang penting. Register menyediakan memory penyangga untuk hasil yang dikeluarkan nanti. Untuk mencapai tujuan ini salah satu masukan pada ALU disambungkan ke keluaran register yang juga merupakan keluaran hasil operasi (feedback). Informasi yang ada di masukan B sekarang dikombinasikan dengan keluaran yang dihasilkan register. Hasil yang dikeluarkan oleh ALU di simpan pada register oleh pulsa clock. Pada tahap ini data yang ada pada register sebelumnya akan hilang. Untuk menghindari kemungkinan munculnya carry yang hanya sesaat saja maka keluaran carry ini disimpan pada suatu flag. Clock pada flag ini berdasarkan bit tambahan S8 dalam ROM (fungsi gerbang AND pada Gambar 12). Hal ini hanya perlu bila carry ini dikeluarkan oleh fungsi ALU. Fungsi yang akan diproses oleh accumulator ini ditampilkan pada tabel 3. Untuk kombinasi U3 sampai dengan U0 = 0101 fungsi operasi keluaran adalah A+1. Karena adanya umpan balik untuk masukan A, maka besarnya A bergantung pada isi yang ada di register saat itu. Dengan instruksi A + 1 ini, isi accumulator ini akan bertambah 1 tiap

satu clock. Dengan kombinasi kontrol seperti ini accumulator disini akan berfungsi sebagai counter.

Jika counter dikehendaki sebagai counter down, maka isntruksi yang harus diberikan adalah dengan mengataur kombinasi U3 sampai dengan U0 = 0110 dan fungsi A -1 akan dilakukan.

Table Fungsi accumulator

U3	U2	U1	U0	Singkatan	Fungsi	Carry Flag
0	0	0	0	NOP	No operation	ya
0	0	0	1	SP1	Set accu = 1	ya
0	0	1	0	CMA	Complement accu	tidak
0	0	1	1	LDA	Load B into accu	tidak
0	1	0	0	CLA	Clear accu	tidak
0	1	0	1	INC	Increment accu	ya
0	1	1	0	DEC	Decrement accu	ya
0	1	1	1	ADD	Add B into accu	ya
1	0	0	0	SUB	Substract B from accu	ya
1	0	0	1	AND	Accu AND B into accu	ya
1	0	1	0	IOR	Accu OR B into accu	ya
1	0	1	1	XOR	Accu XOR B into accu	ya
1	1	0	0	SM1	Set accu = -1	ya
1	1	0	1			
1	1	1	0			
1	1	1	1			

Pada kolom carry flag dapat dilihat kapan carry flag ini keluar pada saat di clock. Pada banyak mikroprosesor flag ini juga di clock untuk operasi logika. Selama tidak ada carry yang dihasilkan oleh ALU, flag ini akan di-clear.

Jika suatu fungsi sangat rumit akan diproses oleh accumulator, maka fungsi ini haraus dibagi menjadi operasi-operasi yang lebih sederhana. Untuk operasi ini diperlukan beberapa siklus.

Sebagai contoh masalah perkalian untuk menghitung 3 B yang mana B adalah data masukan dari saklar. Selama tidak ada fungsi perkalian maka akan diperlukan berepa fungsi yang dasar yang sederhana. Oleh karena itu



kita harus membuat langkah (program) yang harus dilakukan untuk memecahkan masalah perkalian tersebut seperti ditunjukkan beikut ini :

Urutan langkah	Instruksi	Keterangan
1	LDA	Isi accumulator dengan data dari masukan B
2	ADD	Tambahkan isi accumulator dengan data dari masukan B
3	ADD	Tambahkan isi accumulator dengan data dari masukan B

Dalam prakteknya implementasi program di atas adalah sebagai berikut :

1. Siapkan data masukan B
2. Siapkan data kontrol LDA dengan mengatur U3=0, U2=0, U1=1 dan U0=1
3. Beri pulsa clock
4. Siapkan data kontrol ADD dengan mengatur U3=0, U2=1, U1= dan U0=
5. Beri pulsa clock
6. Siapkan data kontrol ADD dengan mengatur U3=0, U2=1, U1=1 dan U0=1
7. Beri pulsa clock

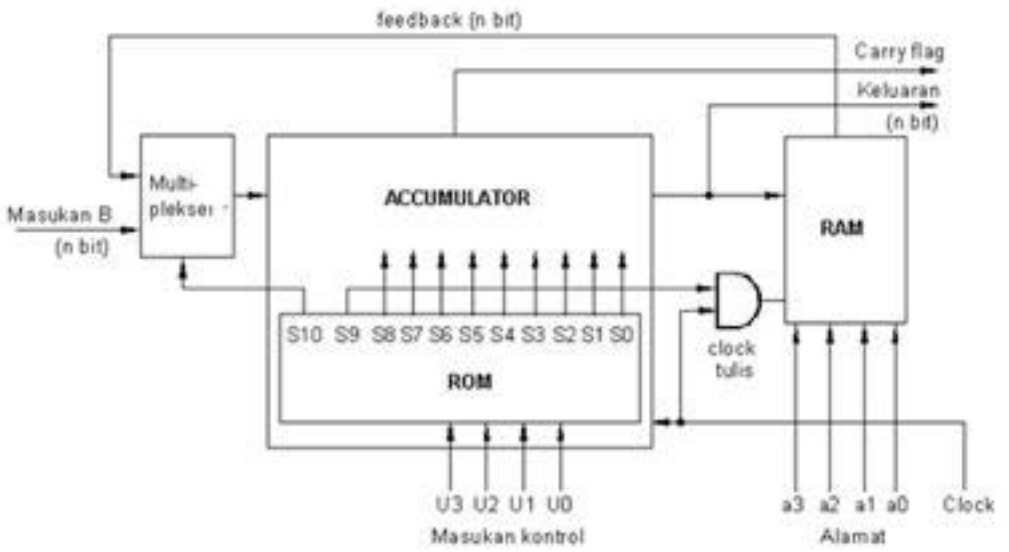
Instruksi pertama LDA menyebabkan data masukan B masuk ke akumulator dengan pulsa clock. Instruksi kedua ADD setelah pulsa clock diberikan akan menghasilkan isi accu sebelumnya ditambah masukan B.

Sekarang B sudah ditambahkan pada isi Accu. B + B akan terbentuk. Dengan fungsi kontrol yang sama, pemberian pulsa clock selanjutnya dibutuhkan untuk menambah B sekali lagi untuk hasil B+B. Sehingga Accumulator akan menghasilkan nilai 3 · B. Dengan memilih kombinasi instruksi kontrol yang tepat, maka ekspresi yang sulit akan dapat dihitung.

2.2.6. Accumulator dengan memory data

Agar supaya accumulator dapat diterapkan pada komputer, diperlukan kemampuan untuk menyimpan hasil operasi kemudian mengambil kembali

untuk pemrosesan data selanjutnya. Kemampuan menyimpan data dan mengambil simpanan data sebelumnya dapat dilakukan oleh sebuah komponen tambahan yaitimemory data atau disebut RAM (Rnadom Acces Memory) yaitu suatu jenis memory yang dapat ditulisi maupun dibaca.



Gambar 2.48. Accumulator dengan memory data

Input B dan output RAM dikontrol oleh multiplekser. Accumulator memiliki keluaran tidak hanya untuk keperluan diluar sistim tetapi juga memiliki keluaran yang digunakan sebagai masukan dari RAM. Dengan demikian memungkinkan untuk mengeluarkan isi akumulator ke memory data atau memasukkan data dari memory melalui multiplekser menuju ke accumulator. Untuk itu perlu penambahan unit kontrol 2 bit yaitu S9 dan S10 untuk melaksanakan instruksi baca tulis memory data. Selanjutnya kode instruksi pada ROM juga berubah dengan tambahan fungsi seperti yang ditunjukkan oleh tabel 3.4.

Keluaran ROM tambahan S10 disediakan sebagai switching multiplekser. Pada kondisi normal multiplekser menghubungkan data memory dengan masukan accumulator.

Dengan kombinasi bit kontrol U3 sampai dengan U0 = 1101, multiplekser membuat masukan B terhubung secara langsung ke accumulator, dimana data akan disimpan sementara oleh register.



Pada saat kombinasi bit kontrol U3 sampai dengan U0 = 0011, multiplekser membuat masukan accumulator terhubung dengan data memory melalui multiplekser.

Bit tambahan S9 pada keluaran ROM digunakan menyiapkan pulsa clock untuk data memory pada saat instruksi menyimpan data ke memory dengan kombinasi bit kontrol U3 ampai dengan U0 = 1110. Untuk fungsi kontrol ini, isi akumuluator akan ditulis ke data memory.

Table Fungsi accumulator dengan memory data

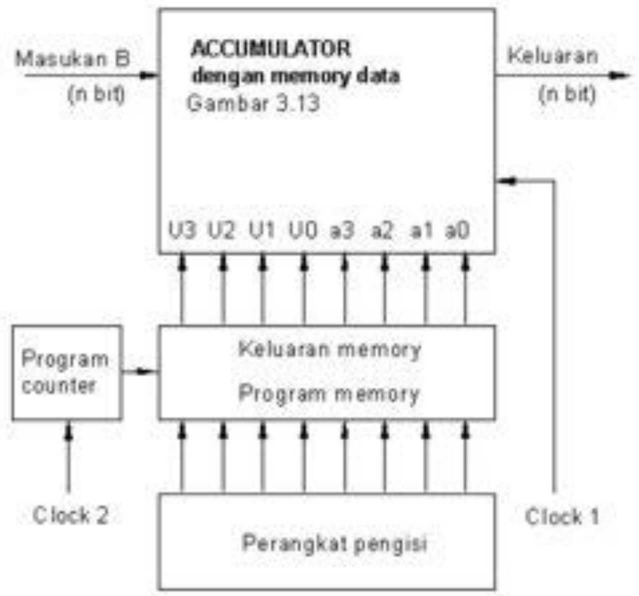
U3	U2	U1	U0	a3	a2	a1	a0	Singkatan	Fungsi	Carry Flag
0	0	0	0	x	x	x	x	NOP	No operation	ya
0	0	0	1	x	x	x	x	SP!	Set accu = 1	ya
0	0	1	0	x	x	x	x	CMA	Complement accu	tidak
0	0	1	1	a	a	a	a	LDA	Load contents address a a a a into accu	tidak
0	1	0	0	x	x	x	x	CLA	Clear accu	tidak
0	1	0	1	x	x	x	x	INC	Increment accu	ya
0	1	1	0	x	x	x	x	DEC	Decrement accu	ya
0	1	1	1	a	a	a	a	ADD	Add contents address a a a a into accu	ya
1	0	0	0	a	a	a	a	SUB	Substract contents address a a a a from accu	ya
1	0	0	1	a	a	a	a	AND	Accu AND contents address a a a a	ya
1	0	1	0	a	a	a	a	IOR	Accu OR contents address a a a a	ya
1	0	1	1	a	a	a	a	XOR	Accu XOR contents address a a a a	ya
1	1	0	0	x	x	x	x	SM!	Set accu = -1	ya
1	1	0	1	x	x	x	x	INP	Load B inputs into accu	tidak
1	1	1	0	a	a	a	a	STA	Store accu into address a a a a	tidak
1	1	1	1	x	x	x	x			

Keterangan : a a a a = alamat memory data

x x x x = tidak dipedulikan

2.2.7. Komputer sederhana

Langkah berikutnya untuk membangun sebuah komputer lengkap adalah dengan menempatkan pola urutan kontrol ke dalam sebuah memory program yang akhirnya memungkinkan operasi secara otomatis dapat dilakukan. Urutan kontrol word atau instruksi (program) yang akan diproses disimpan pada memory program. Urutan program atau instruksi di sini sangat penting. Instruksi yang tersimpan dalam memory harus disimpan secara berurutan sesuai dengan kenaikan alamat yang cesara otomatis akan dilakukan oleh program counter.



Gambar 2.49. Aplikasi program counter dan program memory

Pada intinya bahwa setiap program yang akan dijalankan oleh suatu komputer maka program tersebut harus diisikan ke dalam memory. Ada dua kemungkinan memasukkan program ke dalam memori. Kemungkinan pertama jika program tersebut bersifat tetap dan tidak ada perubahan lagi maka program sejenis ini sebaiknnya disimpan pada ROM (Read Only Memory). Kemungkinan kedua apabila tersebut bersifat sementara atau masih

memungkinkan adanya perubahan perubahan maka sebaiknya program tersebut disimpan pada RAM (Random Acces Memory). Untuk keperluan ini diperlukan perangkat tambahan untuk mengisikan program ke dalam memory.

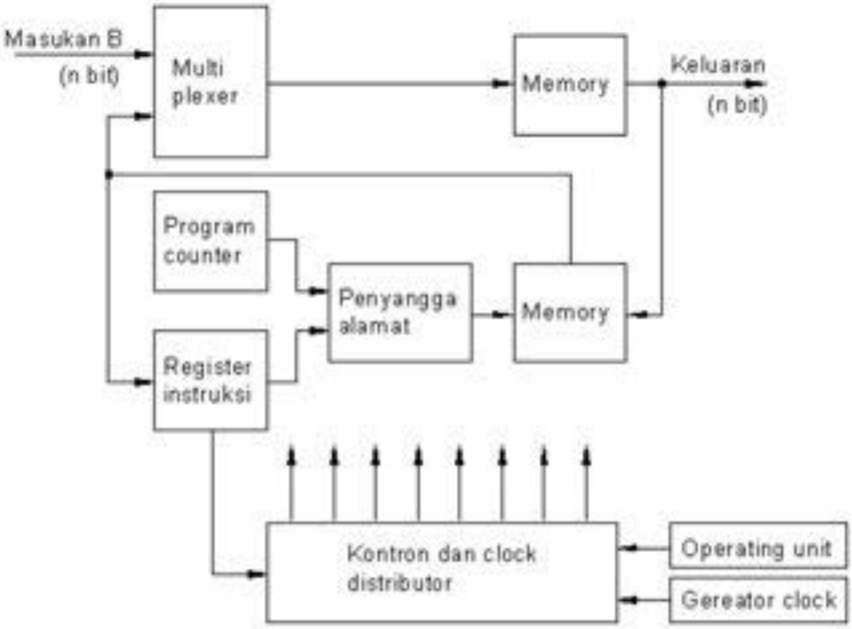
Meskipun beberapa mikroprosesor memisahkan memory tempat menyimpan data dengan memory tempat menyimpan program instruksi, tetapi pada dasrnya suatu memory dapat menyimpan keduanya dalam satu jenis memory bersama. Alternatif lain dalam penerapannya adalah banyak dilakukan dengan menggabungkan ROM dan RAM.

Penggunaan memory bersama (untuk program dan data sekaligus) memiliki keuntungan sebagaia berikut :

- Lebih fleksibel, tergantung dari masalahnya, besarnya memory yang diperlukan untuk data atau untuk program dapat disesuaikan oleh si pemakai apakah program membutuhkan memori lebih banyak atau sebaliknya data memerlukan memory yang lebih besar dari pada program.
- Lebih sedikit sambungan, tidak perlu menyediakan sambungan banyak jenis memory.
- Langkah-langkah program dapat juga diproses sebagai data.

Sedangkan kekurangan dari single memory ini adalah dibutuhkannya rangkaian yang lebih dalam mikroprosesor yang memungkinkan mengambil dan menyimpan data dari program counter atau dari saluran alamat dari suatu instruksi yang semuanya itu harus memlaui suatu multiplekser yang terkontrol. Implementasi yang mungkin dari sistim

komputer tersebut ditujukkan pada Gambar 15. program. data atau untuk program dapat disesuaikan oleh si pemakai apakah program membutuhkan memori lebih banyak atau sebaliknya data memerlukan memory yang lebih besar dari pada programgram ke dalam memori. Kemungkinan pertama jika program tersebut bersifat tetap dan tidak ada perubahan lagi.



Gambar 2.50. Komputer sederhana dengan common data dan program memory

Karena program dalam komputer dapat berjalan secara otomatis, maka diperlukan instruksi HALT untuk menghentikan eksekusi pada saat akhir program. Tanpa instruksi ini maka program akan berjalan tanpa henti. Pola kontrol instruksi HALT adalah dengan mengatur masukan kontrol U3 sampai U0 = 1111.

Control unit dan clock generator diperlukan untuk mengatur langkah-langkah pengoperasian komputer yang bekerja untuk menghasilkan urutan langkah – langkah yang harus dikerjakan oleh tiap – tiap blok bagian komputer.

Program counter memberikan alamat awal program yang akan dijalankan dan sekaligus menaikan satu alamat berikutnya setelah satu instrukasi dijalankan. Isi program counter dikirim ke alamat memory. Instruksi yang ada dimemory dijalankan dan disangga oleh register instruksi.

Instruksi umumnya terdiri dari kode operasi (opcode) dan alamat, kontrol unit akan mengerti apakah suatu instruksi memerlukan mengakses alamat memory atau tidak selanjutnya sinyal kontrol akan didistribusi ke blok-blok yang berkepentingan.



2.2.8. Komputer lengkap

Setiap mikroprosesor selalu dilengkapi dengan instruksi yang dipergunakan untuk mengontrol aliran program. Instruksi penting dalam komputer adalah instruksi loncat (JUMP). Instruksi JUMP adalah memberikan nilai terntu ke program counter. Pada kondisi normal program counter akan menghitung naik satu demi satu, tetapi dengan adanya masukan nilai tertentu ke dalam program counter dapat melakukan perhitungan tidak lagi naik satu demi satu tetapi dapat langsung loncat ke angka tertentu baik pada arah naik maupun mundur. Dengan demikian memungkinkan suatu program diulang-ulang. Instruksi JUMP ini banyak macamnya, lompat dengan syarat atau loncat tanpa syarat ke alamat tertentu pada RAM.

Loncat dengan syarat biasanya syaratnya adalah flag hasil operasi sebelum-nya. Beberapa flag dalam sistim mikroprosesor adalah :

- Carry flag atau C flag, Flag ini mengindikasikan terjadinya overflow hasil operasi melebihi batas bilangan aritmetika integer
- Arithmetic arry flag atau V flag, Flag ini mengindikasikan terjadinya overflow hasil operasi melebihi batas bilangan dalam artimetika komplemen dua
- Zero flag atau Z flag, Flag ini mengindikasikan hasil operasi sama dengan nol
- Negative flag atau N flag, Flag ini mengindikasikan hasil operasi sama dengan negatip
- Parity flag atau P flag, Flag ini mengindikasikan hasil operasi merupakan bilangan ganjil atau genap

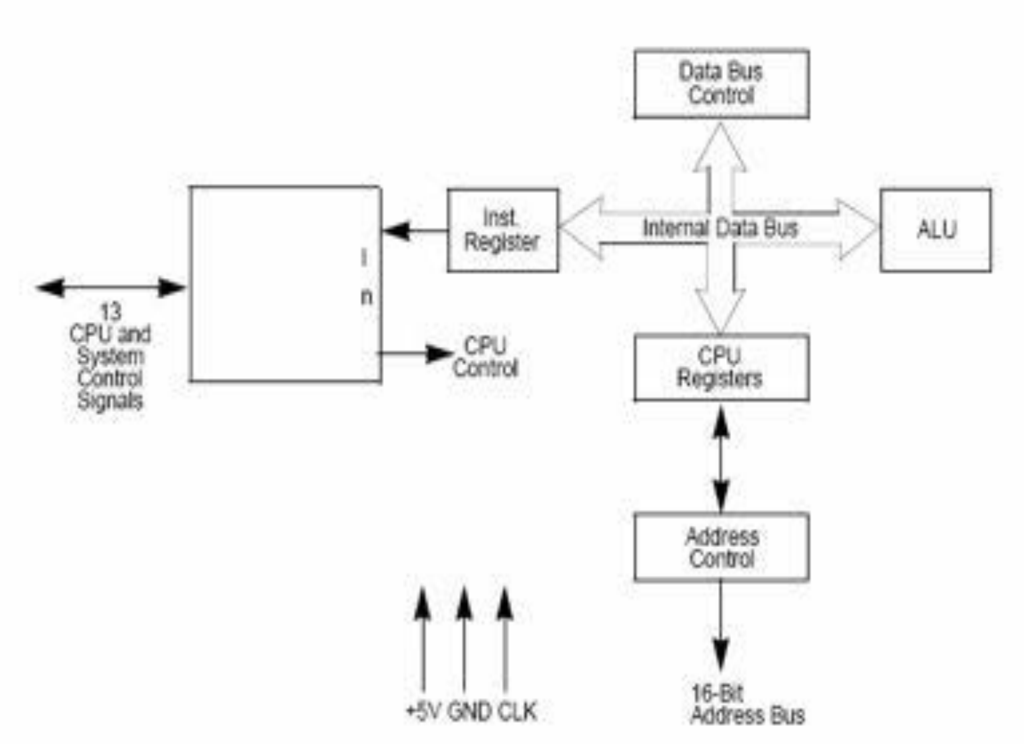


BAB III. SISTEM MIKROPROSESOR

3.1. ARSITEKTUR MIKROPROSESOR Z80

Keluarga CPU Z80 merupakan generasi keempat dari perkembangan teknologi mikroprosesor tnpa kecuali daya komputasi, dan mikroprosesor Z80 ini merupakan sistem dengan teknologi yang cukup tinggi yang memiliki utilisasi memori dengan efisiensi tinggi jika dibandingkan dengan generasi sebelumnya. Kecepatan proses ZILOG CPU Z80 berkisar antara 6 MHz sampai 20 MHz sehingga dengan dukungan software sudah dapat dibentuk sebuah sistem komputer yang cukup handal dan merupakan awal kebangkitan perkembangan komputer. Register internal Z80 yang merupakan memori 208 bit sehingga dapat diakses oleh programer sebagai sebuah fasilitas dalam mengembangkan software aplikasi, register tersebut terdiri dari 2(dua) set dari 6(enam) register general purpose dimana masing-masing register merupakan register individual 8(delapan) atau register pasangan 16 (enam belas) bit. Sebagai register tambahan dilengkapi 2 set register sebagai akumulator dan flag.

Fasilitas lain bagi programer yang diberikan dalam Z80 adalah *Stack Pointer, Program Counter*, 2(dua) register indeks, *REFRESH* register, dan register *INTERRUPT*, sistem sumber daya listrik pada Z80 hanya dibutuhkan sumber tegangan +5V. Seluruh sinyal keluaran secara penuh dikodekan dan pewaktuan guna kepentingan standar kontrol memori atau rangkaian peripheral, CPU Z80 dilengkapi pula dengan ekstensif kontrol peripheral. Gambaran umum elemen dari CPU Z80.



Gambar 3.1. Blok Diagram CPU Z80

3.2. ARSITEKTUR SISTEM MIKROPROSESSOR Z80

Z80 merupakan prosesor 8-bit mikroprosesor yang dirilis bulan Juli tahun 1976 dengan kecepatan clock 2,5 MHz, pada saat itu Z80 lebih berkembang dibanding Intel 8080 begitu juga dibanding Intel 8085. Secara arsitektur Z80 memiliki kesamaan dalam jumlah bit dengan Intel 8080 yaitu untuk jalur data menggunakan 8-bit dan jalur penunjukan alamat baik port I/O maupun alamat memori menggunakan 16-bit. Z80 dapat menjalankan semua op-code Intel 8080 termasuk 80 kode barunya yang meliputi operasi instruksi 1, 4, 8 dan 16-bit dan juga transfer blok dan instruksi blok I/O.

Kelompok register yang digunakan dalam Z80 terdiri dari 2(dua) blok, yang dibagi menjadi 2(dua) bank register termasuk register A dan register F yang dapat saling ditukarkan isi datanya. Dengan demikian sangt memungkinkan penggunaan sistem operasi dengan kecepatan tinggi, atau penggunaan sistem interupsi yang cepat. Untuk sistem interupsi terdapat 3(tiga) buah

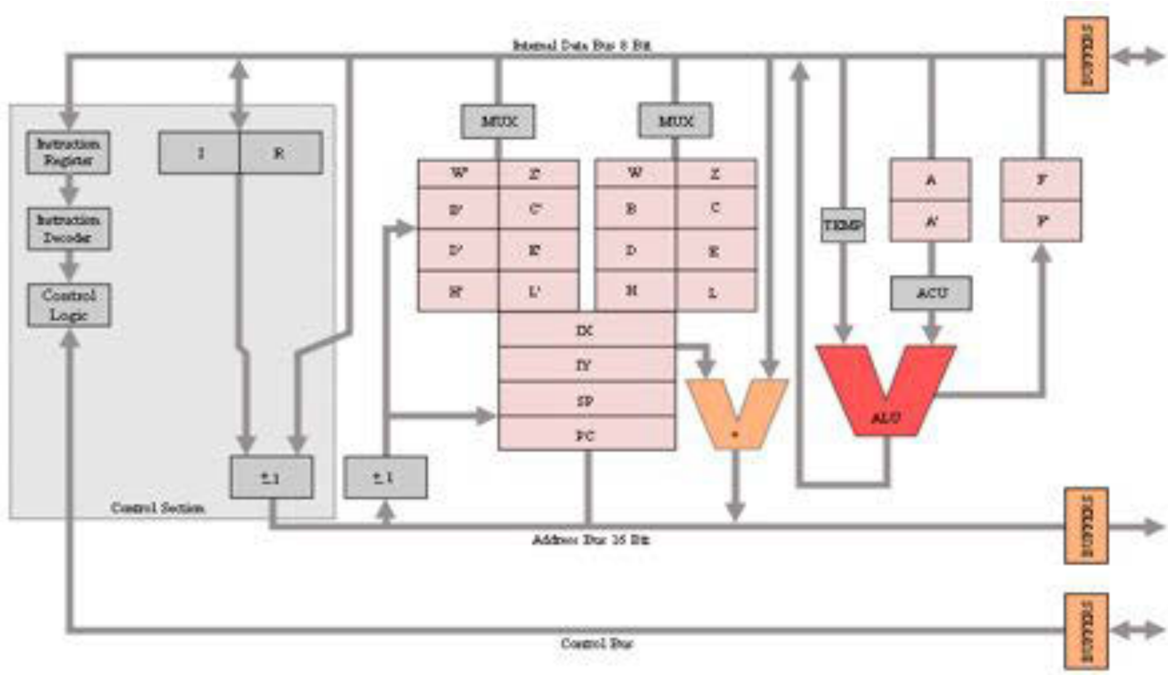
mode, dan pada Z80 juga ditambahkan 2(dua) buah indeks register yaitu register 16-bit IX dan IY serta dilengkapi pula dengan sistem vektor interupsi melalui 8-bit IV register.

Z80 merupakan dasar prosesor dimana interface memori CPU dilengkapi dengan sinyal RAM refresh, sehingga bagi pengembang banyak kemudahan dan harga yang murah. Z80 memiliki kompatibel yang tinggi terhadap 8080 dan CP/M yang merupakan standar sistem operasi mikroprosesor yang pertama.

Terdapat beberapa produksi pilihan yaitu versi original Z80 (2.5 MHz), Zilog Z80A (4 MHz), Zilog Z80B (6MHz) dan Zilog Z80H (8 MHz).

Aplikasi Z80, telah digunakan pada komputer game pada generasi pertama Nintendo Game Boy. Z80 produksi Sharp telah digunakan pada GameBoy Color, running pada clock 4 MHz untuk GameBoy software atau pada 8 MHz untuk Game Boy Color software. Z80 juga sukses digunakan dalam Sega Master System dan Game Gear. Z80 juga digunakan dalam Sega Genesis System untuk hardware reverse yang kompatibel dengan Sega Master System melalui cartridge khusus.

Adapun arsitektur dari mikroprosessor Z80 yang diproduksi oleh Zilog ditunjukan seperti gambar berikut:

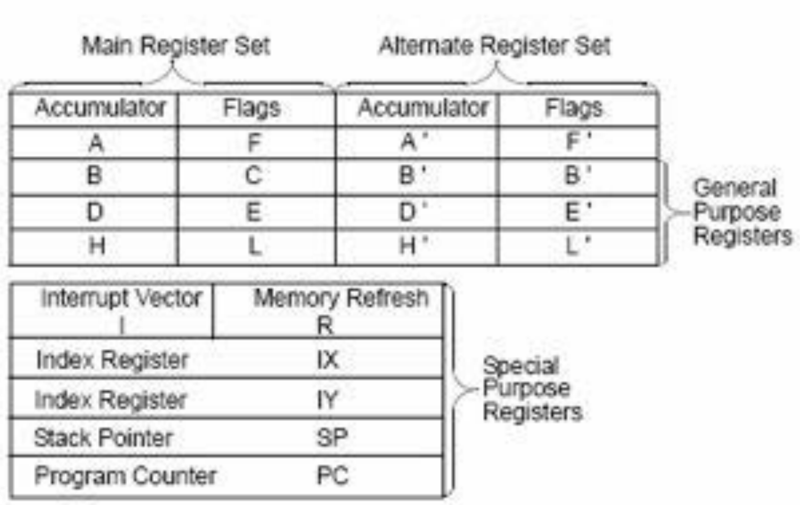


Gambar 3.2. Arsitektur Sistem Mikroprosessor Z80

1. Register Internal CPU Z80

Register internal Z80 yang merupakan memori 208 bit sehingga dapat diakses oleh programer sebagai sebuah fasilitas dalam mengembangkan software aplikasi, gambar 3.3 merupakan ilustrasi konfigurasi tempat penyimpanan bit berupa register yang terdiri dari 2(dua) set dari 6(enam) register general purpose dimana masing-masing register merupakan register individual 8(delapan) atau register pasangan 16 (enam belas) bit. Sebagai register tambahan dilengkapi 2 set register sebagai akumulator dan flag. Semua register dalam mikroprosesor CPU Z80 merupakan implementasi penerapan RAK statik.

Fasilitas lain bagi programer yang diberikan dalam Z80 adalah *Stack Pointer, Program Counter*, 2(dua) register indeks, *REFRESH* register, dan register *INTERRUPT*.



Gambar 3.3. Konfigurasi Register CPU Z80

2. Register Temporer A - F

Register ini merupakan register 8 bit untuk register A dan 8 bit untuk register F, walaupun demikian dalam CPU Z80 kedua register yaitu A dan F dapat diakses secara berpasangan yang berarti merupakan register 16 bit. Fungsi utama untuk register A adalah untuk menampung secara temporer data hasil proses dalam ALU (Arithmatic Logic Unit), sedangkan register F merupakan flag yaitu tanda hasil proses ALU. Untuk memperluas dan kenyamanan programer dalam mengembangkan program aplikasi maka pada CPU Z80 dilengkapi dengan register alternatif A' dan F', adapun fungsi dari register alternatif ini dapat dimanfaatkan seperti halnya pada register utama A dan F.

3. Register Umum (Register pasangan BC, DE, HL)

Register ini dalam program dapat dipakai sebagai register tunggal 8 bit atau sebagai register pasangan 16 bit, yaitu register pasangan BC, DE dan HL. Sehingga saat dipakai sebagai register tunggal maka dia dapat dipakai sebagai penyimpan data 8 bit, dan saat dipakai sebagai register pasangan dia dapat menyimpan data 16 bit. Sebagai contoh alamat lokasi memori yang ingin ditunjuk digunakan register pasangan HL maka jumlah bit penunjuk

alamat adalah 16 bit. Guna mendukung kinerja mikroprosessor dibutuhkan register dan untuk penggunaan register tersebut tersedia perintah khusus operasional register 16 bit.

4. Register Keperluan Khusus (Program Counter (PC))

Register yang diberi nama program counter (PC) ini memiliki kemampuan untuk menunjuk alamat memori 16-bit yang berisi instruksi yang sedang dijalankan dalam memori, PC secara otomatis mengalami inkrimen setelah isinya telah ditransfer ke alamt tertentu, jika terdapat alamat baru yang ditunjukak misal sebagai akibat program harus lompat (jump) maka secara otomatis isi PC akan diubah sesuai dengan alamat yang dituju.

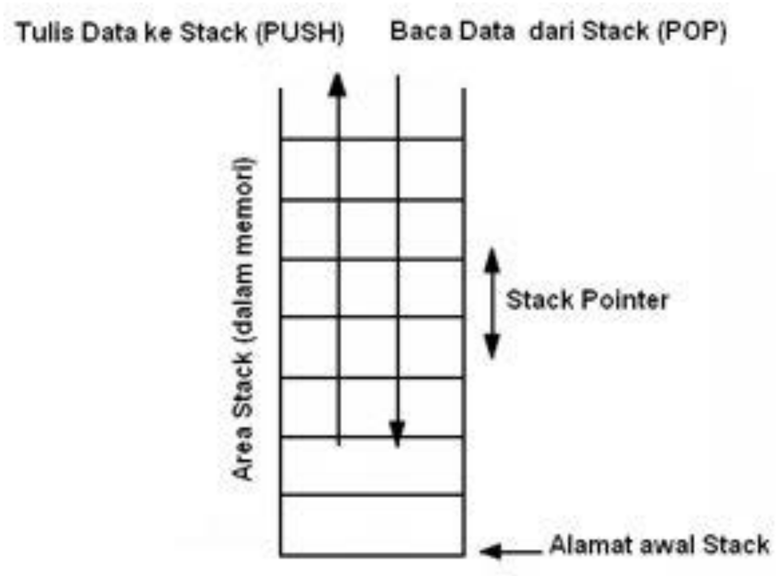
Terdapat 3 kemungkinan PC dalam menjalankan fungsinya yaitu menunjuk alamat memori yang berisi instruksi program, pertama mengikuti urutan sesuai dengan instruksi yang tertulis dalam memori, kedua penunjukan alamat oleh PC yang diset berdasarkan data alamat yang diisikan oleh programer dan yang ketiga dilakukan yang diakibatkan adanya instruksi tertentu (JMP).

5. Stack Pointer (SP)

Register yang diberi nama stack pointer ini memiliki kemampuan untuk menunjuk alamat memori 16-bit berada pada stack puncak yang ditunjuk dan terletak di sembarang tempat dalam sistem RAM. Eksternal memori stack diorganisasi berdasarkan sistem file last-in first-out (LIFO). Data dapat dituliskan (pushed) ke dalam stack dari register tertentu pada CPU atau dibaca ulang (popped) dari stack dari register tertentu pada CPU, instruksi yang digunakan adalah PUSH dan POP. Data yang diambil dari stack melalui instruksi POP selalu diawali dari data yang bealamat terakhir disimpan melalui instruksi PUSH, secara sederhana stack dapat digunakan untuk implementasi in terupsi dengan mode multi level. dan subroutine tak berbatas sistem sarang (nesting) dan penyederhanaan berbagai tipe manipulasi data.

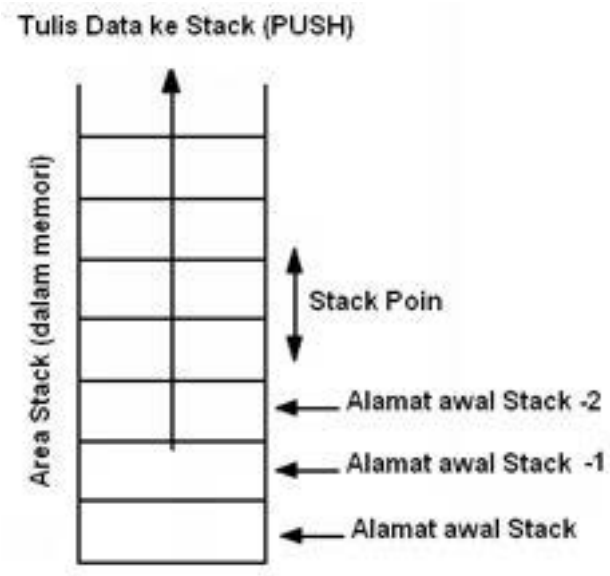
Melalui sebuah program dapat dilakukan akses area stack memori, yaitu dengan menentukan alamat awal stack dilanjutkan penulisan data ke dalam

memori stack pada gambar 3.4 arah penulisan data digambarkan dengan anak panah menjauh dari alamat awal stack. Sedangkan untuk pembacaan data dari stack digambarkan dengan arah panah mendekati alamat awal stack yang ditunjuk.



Gambar 3.4. Area memori stack dan cara aksesnya.

Sebuah contoh, bila sebuah data dari mikroprosessor ditulis ke dalam stack, maka pertama adalah isi dari penunjuk stack dikurangi 1 dan data tersebut di tulis pada alamat ini (alamat awal stack -1), kemudian penunjuk stack dikurangi 1, sehingga data berikutnya ditulis pada alamat awal stack -2. Proses ini terus berlangsung pada setiap penulisan data ke dalam stack. Penunjuk ini terus berlangsung pada setiap penulisan data ke dalam stack, penunjuk stack selalu menunjuk pada alamat lokasi stack yang ditulis terakhir.



Gambar 3.5. Proses Tulis pada Stack

Proses pembacaan data dari stack merupakan metode yang dikenal dengan nama LIFO yaitu singkatan dari Last In First Out. Sehingga pada pembacaan sebuah data dari stack, pertama diawali pembacaan data dari alamat stack yang aktif saat itu (alamat awal stack -2 saat tulis data) dan kemudian penunjuk pointer di tambah 1 dengan demikian stack terisi alamat awal stack -1 yang dapat digunakan untk membaca data berikutnya. Sehingga dari contoh ini dapat disimpulkan bahwa data yang terakhir ditulis pada stack akan di baca pertama pada saat pembacaannya.

6. Dua Indeks Register (IX dan IY)

Kedua register ini merupakan register idependen yang dapat digunakan untuk menunjuk alamat dasar 16 bit (16-bit base address), hal dilakukan melalui mode pengalamat terindeks (indexed addressing mode). Dalam mode ini sebuah register indeks digunakan sebagai penunjuk alamat dasar dalam menunjuk suatu kawasan dalam memori, yaitu alamat dimana data akan dibaca maupun akan disimpan. Tambahan berupa byte diikut sertakan dalam instruksi indeks untuk menspesifikasikan perpindahan dari dasar tersebut. Perpindahan tersebut dispesifikasikan sebagai komplemen dua dari sebuah

integer, hal ini dilakukan untuk tindak penyederhanaan instruksi dalam manipulasi data dalam program terutama dalam penggunaan tabel data.

7. Register dengan Mode Pengalamatan

Mode pengalamatan merupakan pemindahan data dari satu lokasi baik register atau memori untuk diletakan pada register lain atau lokasi memori yang ditunjuk, sedang mode pengalamat dapat dijelaskan sebagai berikut:

a. **Immediate addressing**, mode ini menggunakan bilangan yang merupakan konstanta (immediate) 8 bit sebagai operand, kemudian menyimpannya pada register yang dituju dengan alamat sudah ditentukan. Sintak penulisan instruksi terdiri dari 2(dua) byte), yaitu untuk byte pertama merupakan kode operasi (Opcode) dan byte kedua merupakan data atau konstanta.

   Sebagai contoh menyimpan konstanta 75 dalam bilang hexa ke dalam register A (akumulator), maka penulisan instruksinya secara asembler adalah:

   LD A,75

   dan kode operasionalnya: 3E 75

b. **Immediate extended addressing,** mode ini menggunakan bilangan yang merupakan konstanta (immediate) 16 bit sebagai operand, kemudian menyimpannya pada pasangan register yang dituju dengan alamat sudah ditentukan. Sintak penulisan instruksi terdiri dari 3(tigq) byte), yaitu untuk byte pertama merupakan kode operasi (Opcode) untuk byte kedua dan ketiga merupakan data atau konstanta 16 bit.

   Sebagai contoh menyimpan konstanta 1875 dalam bilang hexa ke dalam register pasangan HL (penunjuk memori), maka penulisan instruksinya secara asembler adalah:

   LD HL,1875

   dan kode operasionalnya: 21 75 18



c. **Relative addressing**, merupakan penunjukan alamat memori yang dilakukan melalui perhitungan relatif. Biasanya digunakan untuk fasilitas lompat (jump) dari lokasi dimana instruksi dilaksanakan menuju lokasi tertentu ditunjuk dengan mode relatif (-128 s/d 127). Instruksi merupakan 2(dua) byte yang terdiri dari kode operasi dan tujuan penempatan relatif (displacement). Adapun contoh instruksi sebagai berikut:

JP 75

dan kode operasionalnya: 18 75

d. **Extended addressing**, mode pengalamatan adalah menunjuk langsung lokasi dimana data berada pada memori untuk disimpan pada sebuah register, adapun penunjuk alamat 16 bit dan kode operasi 1 byte. Contoh instruksi sebagai berikut:

LD A,(1875)

dan kode operasionalnya: 3A 75 18

e. **Indexed addressing**, mode ini menggunakan register index (IX dan IY) untuk menunjuk alamat, operasi pengalamat menggunakan 16 bit. Berikut contoh untuk memindahkan isi register E untuk diletakan pada alamat yang ditunjuk oleh register IX:

LD (IX+08H), E

dan kode operasionalnya: DD 73 08

f. **Register addressing,** mode ini pengalamatan langsung ditunjukan oleh kode operasi pemindahan data dari satu register ke dalam register lainnya. berikut contoh pemindahan isi register D ke dalam Akumulator:

LD A, D

dan kode operasionalnya: 7A

g. **Register Indirect addressing**, mode ini pengalamatan ditujukan untuk memindahkan data yang berada pada register untuk diletakan pada memori dengan alamat yang ditunjuk oleh pasangan register. Sebagai contoh isi akumulator dipindahkan ke memori dengan lokasi yang ditunjuk oleh pasangan register BC:

LD (BC), A

dan kode operasionalnya: 02

8. Interupsi Page Address Register (I)

CPU Z80 dapat digunakan untuk mode operasi dimana pemanggilan langsung pada alamat memori sebagai respon dari pelaksanaan interupsi, register I digunakan untuk tujuan tersebut dan penyimpanan 8 bit high order dari alamat tidak langsung (indirect address) sementara itu peralatan (devais) yang melakukan interupsi melayani 8 bit alamat order rendah. Fitur ini memungkinankan adanya layanan rutin interupsi dinamis yang datanya terletak di berbagai tempat dalam memori dengan waktu akses minimal dalam sebuah rutin. Interupsi secara garis besar dapat dibedakan menjadi dua mode, yaitu:

a. **Mode interupsi non-maskable** merupakan interupsi langsung artinya tidak dihalangi oleh Flip-flop yang berfungsi sebagai interupsi (IFF), layanan diberikan dengan urutan prioritas tertinggi setelah interupsi riset dan Busrq (Bus request).

b. **Mode interupsi maskable** merupakan interupsi yang dihalangi oleh Flip-flop yang berfungsi sebagai interupsi (IFF), layanan mode ini terbagi menjadi 3(tiga) meliputi:

- **mode 0**, yaitu dengan adanya sinyal interupsi ini CPU akan menunjuk salah satu dari alamat 0000H, 0008H, 0010H, 0018H, 0020H, 0028H, 0030H dan 0038H.
- **mode 1**, CPU akan menunjuk alamat 0038H setelah menerima sinyal interupsi.
- **mode 2**, merupakan interupsi yang terkait dengan port I/O, CTC, SIO dan DMA dengan penunjukan alamat dilakukan oleh register I untuk 8 bit atas dan 8 bit bawah ditunjuk oleh peminta interupsi, untuk alamat yang ditunjuk merupakan satu area memori (page) yang terdiri dari 128 lokasi.

9. Arithmetic Logic Unit (ALU)

Instruksi 8 bit unrul aritmatika dan logik dari CPU Z80 dijalankan dalam ALU, dan pada proses ini ALU berkomunikasi dengan register serta data bus eksternal melalui jalur bus data. Fungsi operasional ALU meliputi:

- Penjumlah
- Pengurang
- Logika AND
- LogikaOR
- Logika Exclusive OR
- Perbandingan
- Geser Kiri atau Kanan atau Putar (Arithmetic and Logical)
- Inkrimen
- Dekrimen
- Set Bit
- Reset Bit
- Test bit



10. Instruksi Register dan Kendali CPU

Setiap kali PC menunjuk lokasi instruksi dalam memori untuk dijalankan maka instruksi tersebut diletakan di dalam register instruksi, yang kemudian instruksi tersebut diterjemahkan untuk bisa dilaksanakan oleh CPU (decode). Sebagai hasil pengkodean akan direspon oleh CPU secara hardware, yaitu meliputi pembangkitan sinyal kontrol, baca dan tulis (read and write) data dari dan ke register, kontrol ALU serta layanan sinyal kontrol eksternal divais.

a. Register Perintah

Register perintah diisi langsung dari bus data sistem melalui bus data internal. Pada informasi 8 bit yang dibawah ke register ini adalah selalu menunjukkan suatu kode operasi dari sebuah perintah.

b. Pendekoder Perintah

Masing-masing bit dalam register perintah di uji / di periksa keadaan tegangannya (H atau L) oleh pendekoder perintah . Dengan demikian hal tersebut dapat dipastikan bahwa informasi yang disimpan dalam register perintah adalah merupakan suatu kode operasi tertentu.

c. Pengontrol Waktu dan Aliran (Pengontrol Waktu dan Aliran)

Unit ini berfungsi mengkoordinasikan antara jalannya sinyal di dalam dan di luar mikroprossesor dengan waktu. Unit pengontrol ini menyimpan informasi internal mikroprossesor yang berasal dari pendekoder perintah dan dari luar unit sistem. Sinyal yang di terima dari luar adalah sinyal detak (clock), sinyal control (WR/RD) dan sinyal penawaran (Riset, int) pengontrol waktu dan logik memberikan informasi balik pada unit sistem seperti sinyal tulis diberikan ke unit sistem menunjukkan bahwa pada unit ini akan di tulis sebuah data Keseluruhan dari sinyal masuk dan keluar pada unit pengontrol waktu dan logika ini disebut bus kontrol.

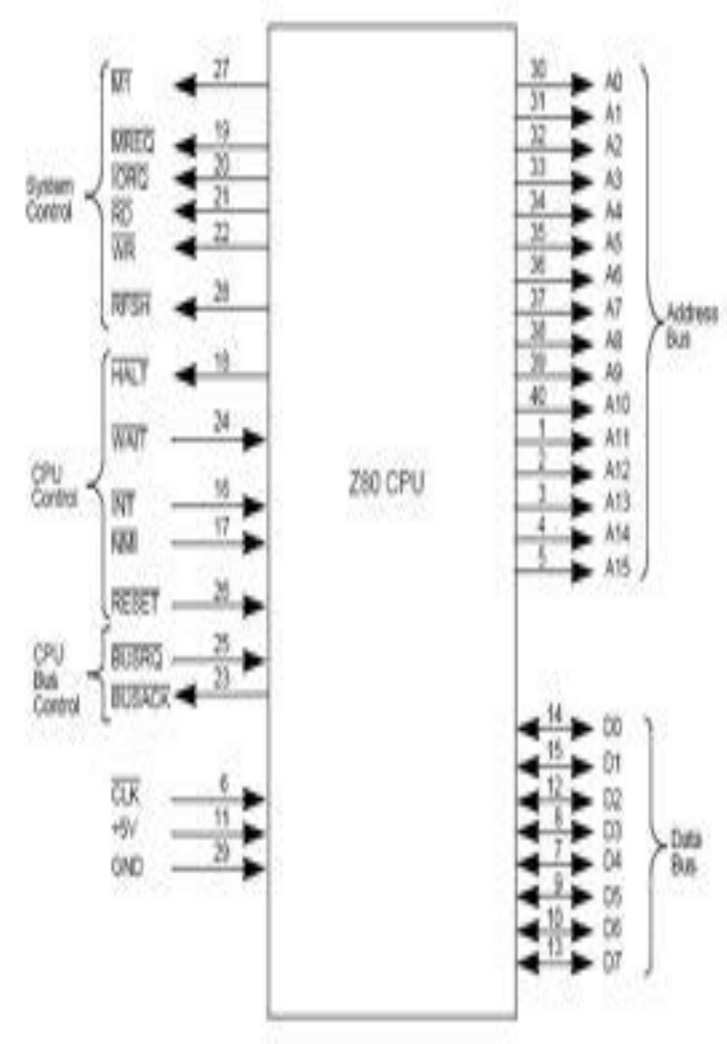
d. Memory Refresh Register (R)

CPU berisi Z80 counter refresh memori, memori dinamis memungkinkan untuk digunakan dengan kemudahan yang sama dengan memori statis. Tujuh bit dari register 8-bit secara otomatis bertambah setelah setiap instruksi

dijalankan. Bit kedelapan sebagai pengingat program, akibat pelaksanaan dari instruksi LD R, A . Data di counter refresh dikirim keluar pada bagian bawah dari bus alamat bersama dengan sinyal kontrol refresh sementara CPU melakukan decoding dan mengeksekusi instruksi. Mode refresh merupakan pemutakhiran isi memori yang penting untuk programmer dan refresh tidak memperlambat operasi CPU. Programmer dapat memuat register R untuk tujuan pengujian, namun register ini biasanya tidak digunakan oleh programmer. Selama refresh, isi dari register I ditempatkan pada delapan bit atas bus alamat.

3.3. PIN HARDWARE MIKROPROSESSOR Z80

Mikroprosesor Z80 dikemas dalam bentu IC dengan jumlah 40 buah pin, yang fungsinya untuk menghubungkan CPU dengan berbagai divais I/O untuk dipasangkan padanya sebagai pembentuk sistem mikroprosesor. Sebagai ilustrasi diskripsi setiap pin IC Z80 dapat dilihat pada gambar berikut:



Gambar 3.6. Pin IC Mikroprosesor Z80

Tabel Fungsi Pin Z80

Simbol	No. PIN	Fungsi	Diskripsi
Ao ... A15	1 ... ..5 30 ... 40	Out	Tristate output, address bus dapat menentukan alamat memori 64 KByte dan 8 bit terendah untuk menentukan alamat 256 I/O berupa peralatan I/O dalam proses penukaran data. Untuk kebutuhan pengalamatan masukan dan keluaran (I/O) dibutuhkan 8 bit rendah dari CPU

Simbol	No. PIN	Fungsi	Diskripsi
			(A0 ... A7). Sedangkan untuk pengalamatan isi akumulator dibutuhkan 8 bit tinggi (A8 ...A15). Pada pengalamatan port juga menggunakan sinyal dari alamat A8 ... A15.
Do ... D7	7, 8, 9, 10, 12, 13,14,15	Inp/Out	Tristate input/output, merupakan 8 bit data bus dua arah dan berfungsi untuk melayani proses transfer data.
INT	16	Inp	Input aktip berlogika 0, interup ini dihasilkan oleh peralatan I/O. Jika CPU menerima interup INT maka signal IORQ selama waktu MI akan dikeluarkan CPU pada awal siklus instruksi berikutnya.
NMI	17	Inp	Input triger /negatip, mempunyai prioritas lebih tinggi dari INT dan signal ini akan menempatkan PC pada alamat 0066 H dan secara otomatis menyimpan isi PC pada stack sehingga setelah terjadi interupsi ini pemrogram dapat mengalihkan ke proses program sebelum diinterup.
HALT	18	Out	Signal LOW pada HALT memberitahukan bahwa CPU telah melaksanakan instruksi HALT dan sekarang menunggu Interupt. Selama keadaan HALT, CPU menyelesaikan instruksi NOP untuk mempertahankan refresh. Nop = No Operation HALT = Penghentian
MREQ	19	Out	Tristate output aktip dengan logika 0, untuk melayani permintaan proses transfer data yang



Simbol	No. PIN	Fungsi	Diskripsi
			menggunakan memori
IORQ	20	Out	Tristate output aktip dengan logika 0, untuk melayani permintaan proses transfer data yang menggunakan I / O
RD	21	Out	Tristate output aktip dengan logika 0, merupakan signal yang dikeluarkan oleh CPU jika ingin membaca data baik dari memori maupun dari I / O
WR	22	Out	Tristate output aktip dengan logika 0, merupakan signal yang dikeluarkan oleh CPU jika ingin menulis data baik dari memori maupun dari I / O
BUSAK	23	Out	Output aktip berlogika 0, signal ini memberikan informasi kepada peralatan luar CPU bahwa Adress Bus, data bus dan tristate output signal kendali pada keadaan impedansi tinggi serta siap untuk dikendalikan oleh peralatan luar
WAIT	24	Out	Input aktip berlogika 0, memberikan signal bahwa address memori atau I / O tidak siap untuk proses data transfer dan CPU akan aktip kembali jika signal wait aktip.
BUSRQ	25	Out	Input aktip berlogika 0, signal ini meminta CPU agar address bus, data bus dan tristate output signal kendali pada keadaan impedansi tinggi sehingga memungkinkan peralatan lain dapat mengendalikan bus - bus tersebut.

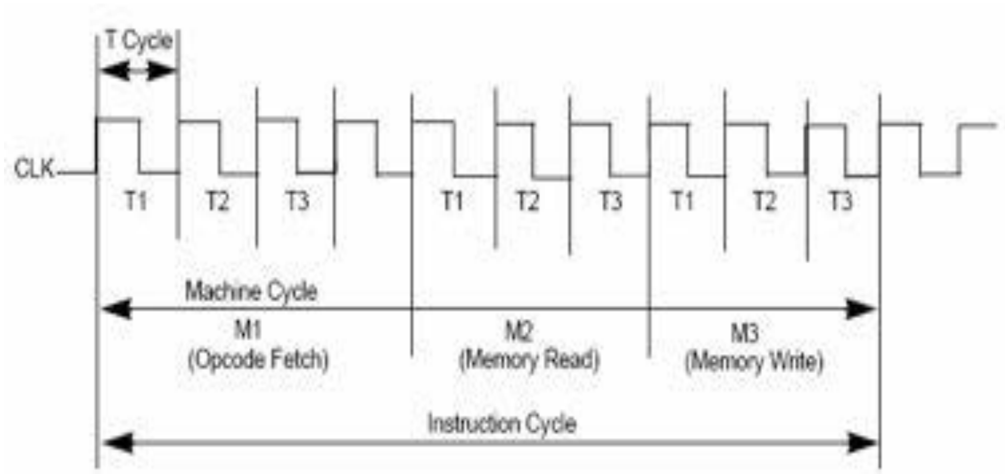
Simbol	No. PIN	Fungsi	Diskripsi
RESET	26	Inp	menempatkan isi PC = 00H, register I = 00H, register R = 00H dan Interupt Mode = 0. Selama waktu reset address bus dan data bus mempunyai impedansi tinggi dan output signal kendali pada keadaan tidak aktip.
M1	27	Out	Output aktip dengan logika 0, memberikan signal indikasi pelaksanaan op-code instruksi selama satu siklus mesin, untuk 2 byte op-code akan dihasilkan signal setiap satu siklus
RFSH	28	Out	Output aktip berlogika 0, menunjukkan bahwa 7 bit terendah dari address bus berisi refresh addres memori dinamis dan bersama signal MREQ untuk membaca memori dinamis.

3.4. Sistem Timing

Dalam operasi CPU pada Z80 dirancang oleh Zilog dalam mengeksekusi instruksi melaui tahapan yang sangat presisi, termasuk untuk operasi:

- Baca/tulis Memori (R/W)
- Baca/tulis devais I/O (R/W)
- Layanan sistem Interupsi

Secara keseluruhan instruksi merupakan sederetan operasi dasar Z80, setiap operasi memerlukan tiga sampai enam periode clock atau lebih untuk sikronisasi CPU terkait dengan kecepatan akses devais eksternal. Periode clock ditentukan berdasarkan waktu siklus T(time) dan operasi berdasarkan siklus M (mesin).



Gambar 3.7. Dasar Timing Operasi CPU

Satu periode clock dinyatakan sebagai satu siklus (T), setiap operasi instruksi dinyatakan sebagai siklus instruksi dan setiap siklus instruksi terdiri dari tiga siklus mesin (M1, M2, M3). Siklus mesin pertama M1 untuk sembarang instruksi merupakan siklus fetch yang dilaksanakan melalui empat, lima atau enam siklus T, yaitu sampai muncul sinyal wait yang diteruskan instruksi berikutnya untuk dilaksanakan. Sedangkan yang dimaksud dengan siklus fetch adalah siklus eksekusi instruksi dalam periode waktu dimana mikroprosesor membaca dan memproses instruksi dari memori, dan mengeksekusinya. Proses ini merupakan siklus yang berkelanjutan, dijalankan terus sampai mikroprosesor dimatikan atau ada permintaan untuk proses lain. Adapun siklus mesin merupakan operasi pemindahan data antara CPU dengan memori atau devais I/O, untuk itu dibutuhkan clock sebanyak tiga sampai lima clock.

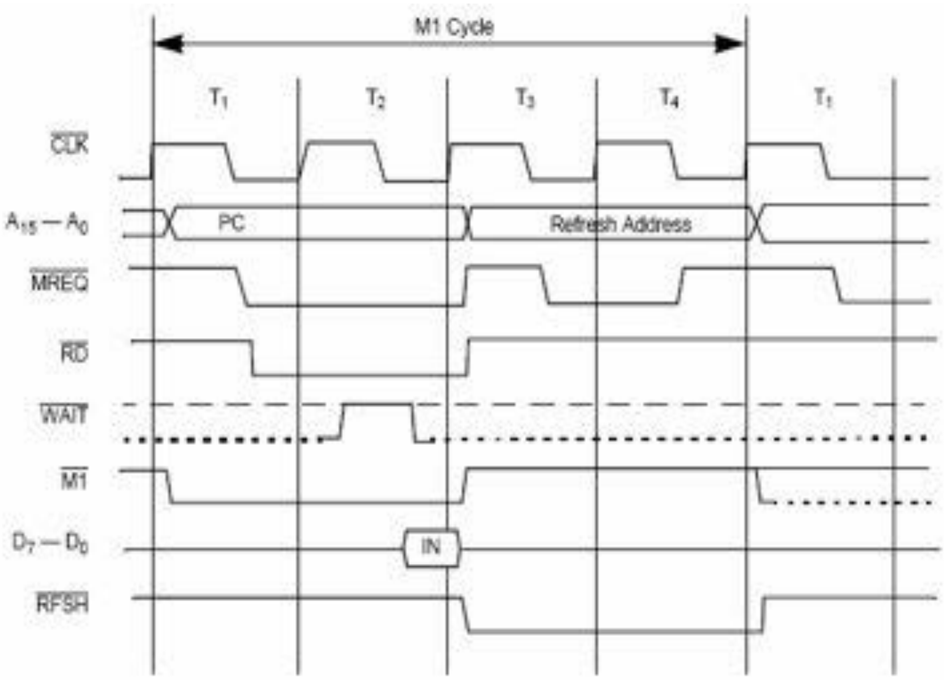
3.5. FETCH INSTRUKSI

Gambar 2.58 menggambarkan timing selama siklus (opcode fetch) M1. Isi PC ditempatkan pada bus alamat pada saat awal siklus M1. Satu setengah siklus clock berikutnya sinyal $\overline{MREQ}$ akan aktif. Pada saat ini pengalamat memori memiliki waktu stabilisasi sehingga tepi turun (falling edge) $\overline{MREQ}$ dapat

digunakan secara langsung sebagai clock membuka chip (chip enable) untuk memori dinamis. Jalur RD menjadi aktif yang berarti proses baca data data memori dilakukan sekaligus data diletakan pada bus data, dan pembacaan data oleh CPU dari memori ke bus data dilakukan saat tepi naik clock state T3, bersamaan dengan itu digunakan oleh CPU untuk mematikan sinyal RD dan $\overline{MREQ}$.

Jadi data telah dibaca oleh CPU sebelum sinyal RD menjadi in-aktif, selanjutnya clock state

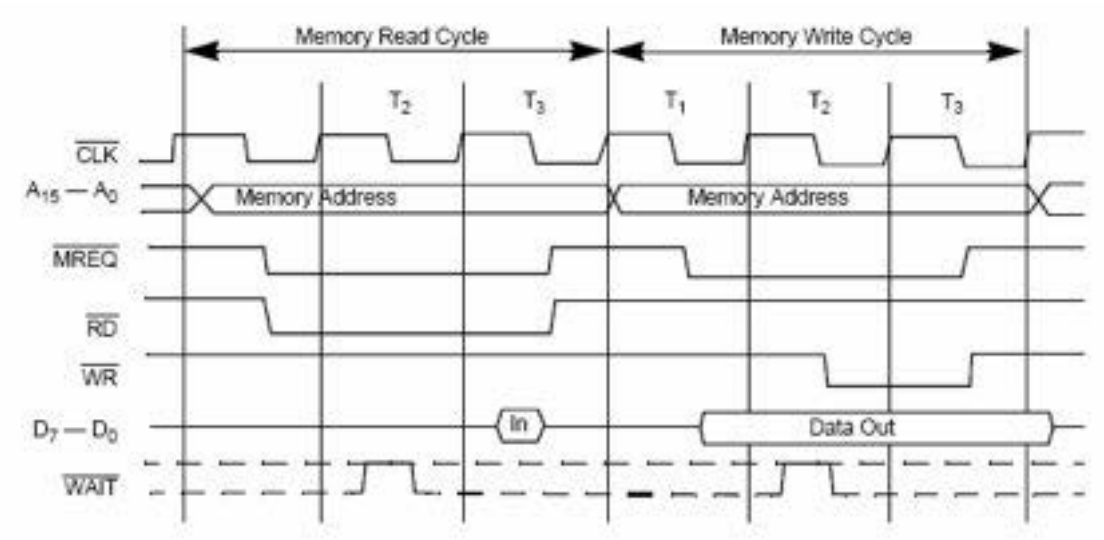
T3 dan T4 dari siklus fetch digunakan untuk proses refresh memori dinamik. Saat ini waktu digunakan oleh CPU untuk mendekode dan mengeksekusi instruksi fetch, sehingga tidak terjadi operasi lain pada saat ini. Selama T3 dan T4, bit tujuh bawah dari bus alamat berisi memori alamat refresh dan sinyal RFSH menjadi aktif, hal ini mengindikasikan bahwarefresh baca semua memori dinamik sudah harus terselesaikan. Sinyal RD tidak dimunculkan selama waktu refresh untuk menjaga data dari segmen memori lain agar tidak masuk bus data. Sinyal $\overline{MREQ}$ selama waktu refresh digunakan untuk melakukan refresh baca pada semua elemen memori, dan sinyal refresh tersebut tidak dapat digunakan manakala pengalamatan refresh tidak stabil selama waktu $\overline{MREQ}$.



Gambar 3.8. Op Code Fetch Instruksi

3.6. BACA – TULIS MEMORI (R/W)

Gambar 3.9 merupakan ilustrasi waktu siklus baca dan tulis memori, siklus ini secara umum terdiri dari tiga periode clock kecuali state wait diminta oleh memori melalui sinyal WAIT. Sinyal MREQ dan sinyal RD dalam penggunaannya sama seperti dalam siklus fetch, pada saat siklus tulis memori sinyal MREQ menjadi aktif ketika kondisi bus alamat stabil sehingga dapat digunakan untuk membuka chip untuk memori dinamik. Jalur WR akan aktif manakala data pada bus data stabil, sehingga dapat digunakan untuk R/W pada berbagai tipe memori. Sinyal WR akan in-aktif satu setengah state T sebelum pengalamatan dan isi bus data berubah, sehingga tumpang tindih bisa terjadi untuk hampir semua tipe memori .

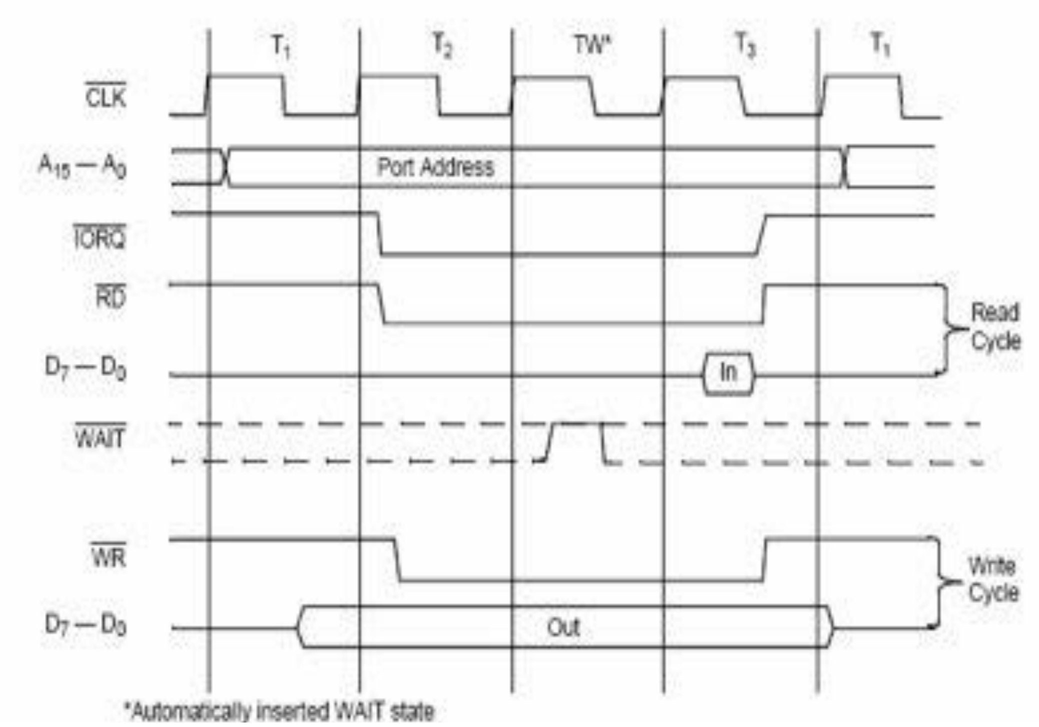


Gambar 3.9. Siklus Baca dan Tulis Memori

3.7. SIKLUS INPUT - OUTPUT

Gambar 3.10 merupakan ilustrasi operasi baca dan tulis jalur I/O, selama operasi I/O state wait tunggal secara otomatis dimasukan. Alasannya adalah selama operasi I/O yaitu saat sinyal $\overline{IORQ}$ aktif sampai CPU menunjuk sinyal $\overline{WAIT}$ (waktunya sangat pendek), tanpa state tambahan maka waktu tidak cukup akibatnya untuk dekode pengalamatan sebuah I/O dibutuhkan aktivasi jalur $\overline{WAIT}$. Jadi tanpa adanya state wait sulit untuk rancangan devais I/O MOS untuk dapat bekerja pada kecepatan penuh CPU, dan selama waktu state wait dibutuhkan sinyal $\overline{WAIT}$.

Selama waktu operasi baca I/O, maka jalur $\overline{RD}$ digunakan untuk melakukan pembukaan pengalamatan port ke bus data, hal ini dilakukan sebagaimana terjadi pada baca memori. Untuk operasi I/O tulis jalur $\overline{WR}$ digunakan sebagai clock ke port I/O.

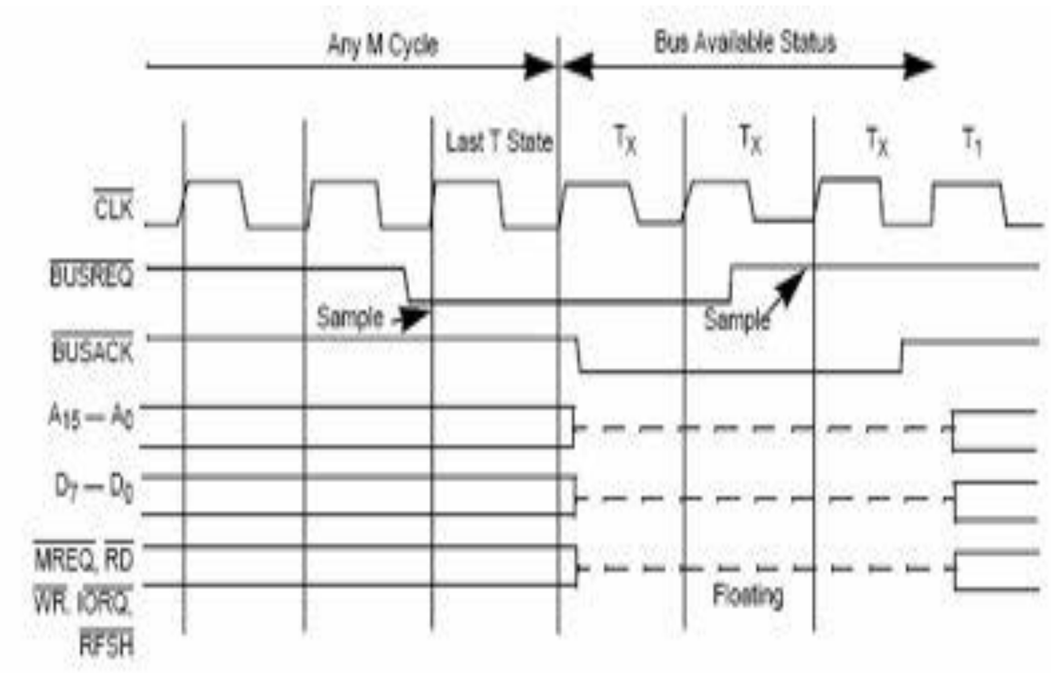


Gambar 3.10. Siklus Operasi Baca-Tulis I/O

3.8. SIKLUS PERMINTAAN BUS

Gambar 3.11 merupakan ilustrasi timing untuk siklus permintaan atau pengakuan bus (Bus Request/Acknowledge), sinyal $\overline{BUSREQ}$ keberadaannya dikenali oleh CPU melalui tepi kenaikan dari periode clock terakhir dari siklus mesin. Pada saat sinyal $\overline{BUSREQ}$ aktif maka CPU akan menset alamat, data dan sinyal kontrol tristate pada state impedansi tinggi yaitu saat tepi naik dari clock pulse berikutnya, pada waktu tersebut devais eksternal dapat mengendalikan bus untuk melalukan transfer antara memori dengan devais I/O. Operasi inilah yang dikenal dengan istilah akses memori secara langsung "Direct Memory Access" [DMA] menggunakan siklus stealing. Adapun waktu maksimum yang dibutuhkan oleh CPU untuk memberikan respon terhadap permintaan bus (bus request) adalah sepanjang siklus mesin, dan untuk pengendalian dari luar dapat dilakukan pengelolaan kontrol terhadap bus sesuai permintaan yaitu sebanyak siklus clocknya.

Jika pemakaian siklus DMA terlalu panjang, dan memori dinamis digunakan maka pengendali eksternal juga melakukan fungsi refresh. Situasi tersebut akan muncul manakala transfer data dalam blok sangat besar dilakukan melalui kendali DMA, dan selama siklus permintaan bus maka CPU tidak dapat diinterupsi oleh sinyal NMI atau sinyal INT.

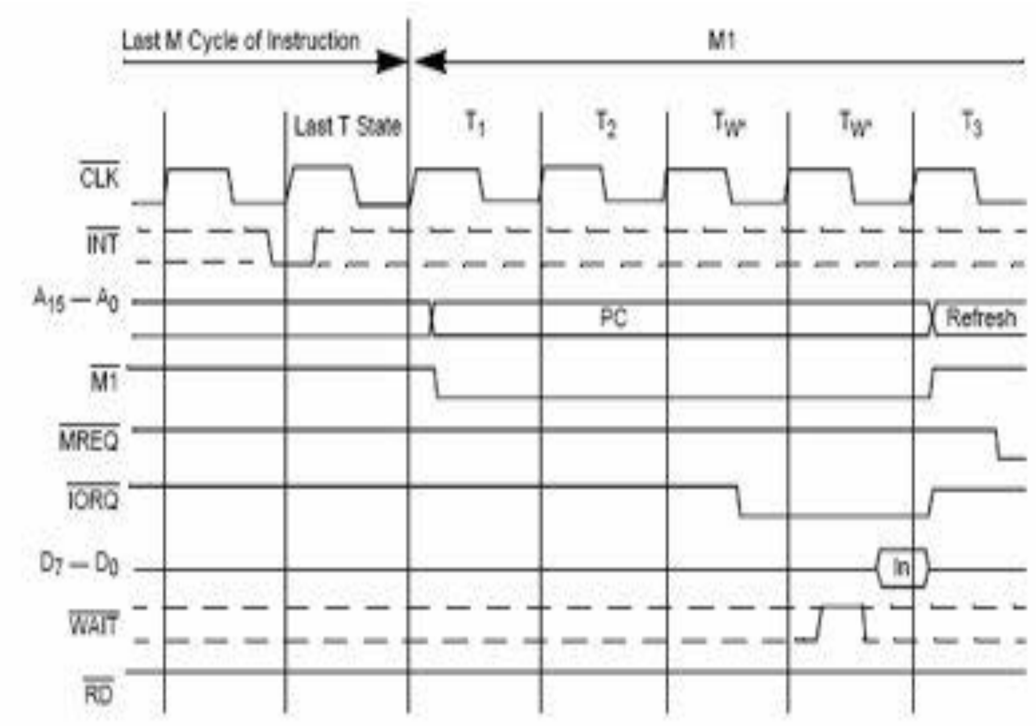


Gambar 3.11. Siklus **Bus Request/Acknowledge**

3.9. SIKLUS INTERUPSI REQUEST/ACKNOWLEDGE

Gambar 3.12. merupakan ilustrasi dari siklus Interupsi, dan CPU membaca sinyal interupsi (INT) pada saat tepi naik clock terakhir dari setiap pelaksanaan instruksi. Sinyal tidak diterima manakala software internal CPU yang mengendalikan interupsi enable flip-flop tidak diset atau ketika sinyal BUSREQ dalam keadaan aktif, jika sinyal diterima maka siklus khusus M1 dimunculkan. Selama siklus khusus M1 maka IORQ menjadi aktif (termasuk pada kondisi normal MREQ), hal tersebut mengidikasikan bahwa devais yang meninterupsi diijinkan untuk menempatkan vektor 8-bit pada bus data. Dua

state wait secara otomatis ditambahkan ke dalam siklus, sehingga ripple interupsi prioritas dapat dengan mudah dapat dilakukan. Dan dengan adanya dua state wait memungkinkan fasilitasi terhadap ketidak cukupan waktu tersedia untuk stabilisasi dan identifikasi sinyal ripple yang harus dilakukan oleh devais I/O memasukan vektor respon.

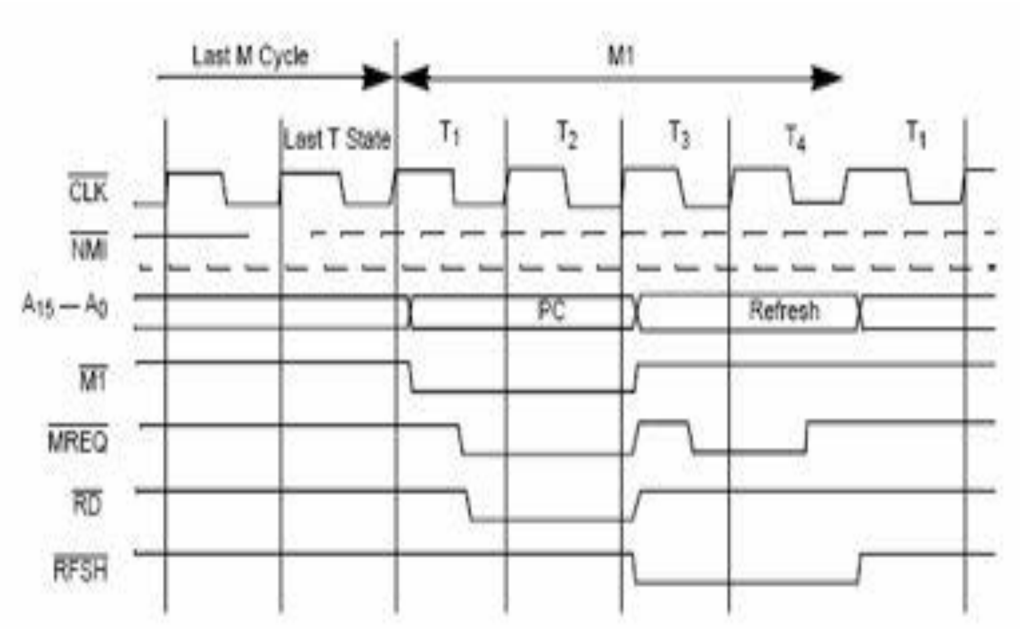


Gambar 3.12. Siklus Interupsi Request/Acknowledge

3.10. Interupsi Non-Maskable

Gambar 3.13 merupakan ilustrasi siklus permintaan interupsi non-maskable, sinyal diambil oleh CPU pada waktu yang bersamaan seperti halnya pada jalur interupsi. Pada interupsi ini merupakan prioritas utama dibanding interupsi normal, interupsi tidak bisa disable dibawah kendali software. Interupsi ini merupakan fungsi khusus untuk layanan respon terhadap sinyal penting seperti adanya kesalahan pada sistem power. Respon CPU terhadap interupsi non-maskable adalah sama seperti halnya operasi normal baca operasi memori, perbedaannya hanya pada isi dari bus data dibuat ignore sementara itu prosesor otomatis menyimpan isi PC ke dalam stack eksternal

dan lompat ke alamat 0066H. Sedangkan rutin layanan interupsi non-maskable dimulai dari lokasi tersebut manakala interupsi digunakan.



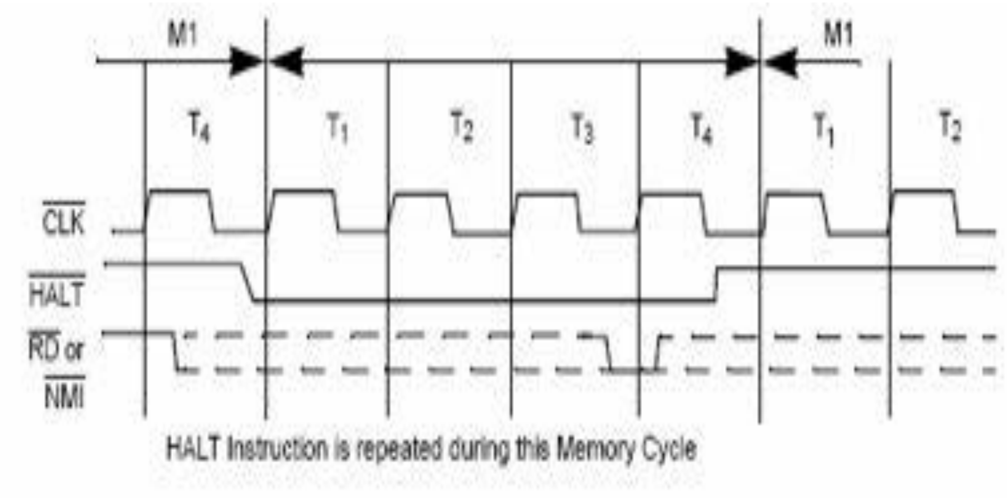
Gambar 3.13. Siklus Interupsi Non-Maskable

3.11. HALT EXIT

Ketika software melaksanakan instruksi HALT maka CPU melaksanakan instruksi NOP sampai adanya insterupsi diterima, dimana instrupsi ini dapat berupa interupsi non-maskable atau maskable selama flip-flop interupsi dalam kondisi enable. Kedua jalur interupsi tersebut dibaca oleh CPU pada saat tepi clock naik yaitu setiap state T4 dijalankan . Jika interup non-maskable atau maskable diterima dan flip-flop interup dalam kondisi diset enable maka keluar dari state HALT dilakukan pada tepi clock naik berikutnya.

Jika kedua sinyal inrupsi tersebut diterima dalam waktu bersamaan maka interup non-maskable maerupakan acknowledge yang mendapat prioritas tertinggi, adapun tujuan dijalankannya instruksi NOP selama pelaksanaan HALT adalah untuk menjaga agar refresh terhadap memori tetap aktif dilakukan.

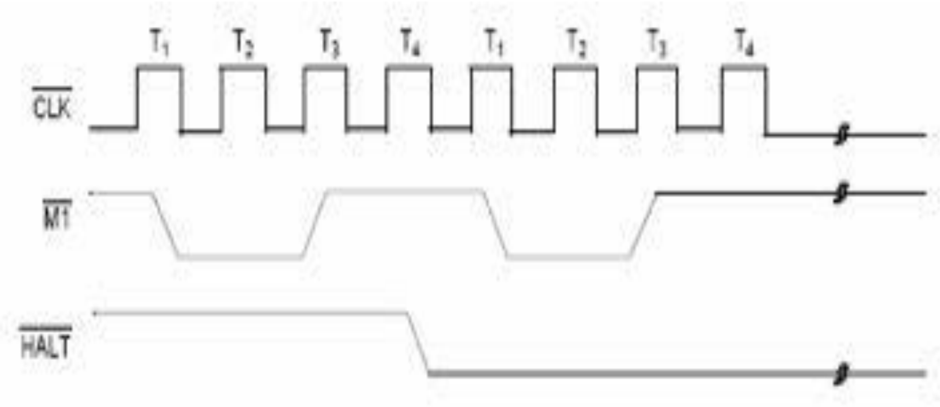
Setiap siklus dalam state HALT merupakan siklus normal M1 (fetch), kecuali untuk penerimaan data dari memori diabaikan (ignore) dan instruksi NOP menekan secara internal ke CPU. Jadi sinyal HALT akan terus aktif selama waktu ini sehingga prosesor menentukan dalam state HALT.



Gambar 3.14. Siklus Pelaksanaan Instruksi HALT

3.12. Siklus Power-Down Acknowledge

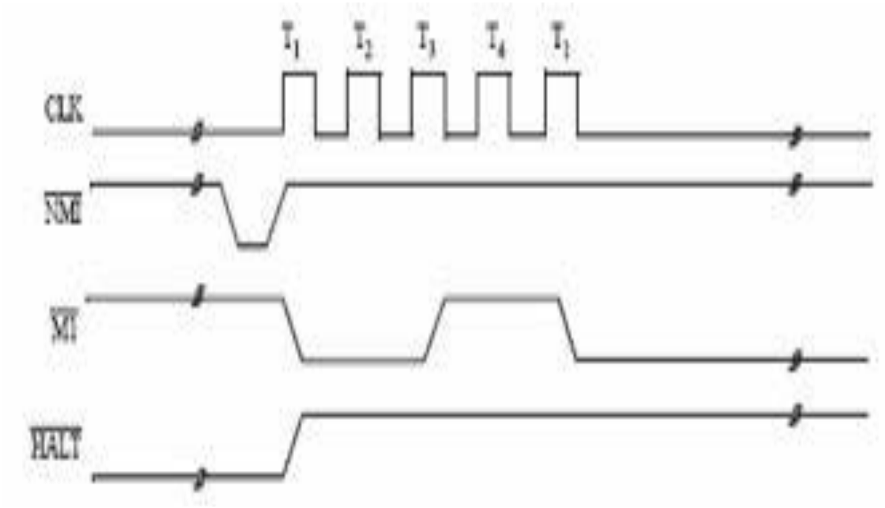
Pada saat input clock ke CPU terhenti, yaitu pada saat kondisi High atau pada kondisi Low maka CPU akan menghentikan operasinya dan pada kondisi ini masih mempertahankan kondisi register dan sinyal kontrol. Berikut merupakan diagram waktu untuk siklus fungsi power-down, dimana instruksi HALT dijalankan.



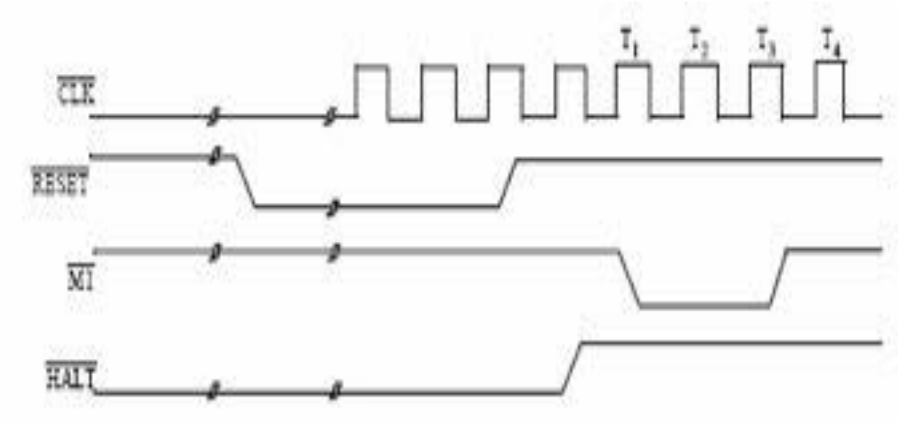
Gambar 3.15. Siklus Power-Down Acknowledge

3.13. Siklus Power-Down Release

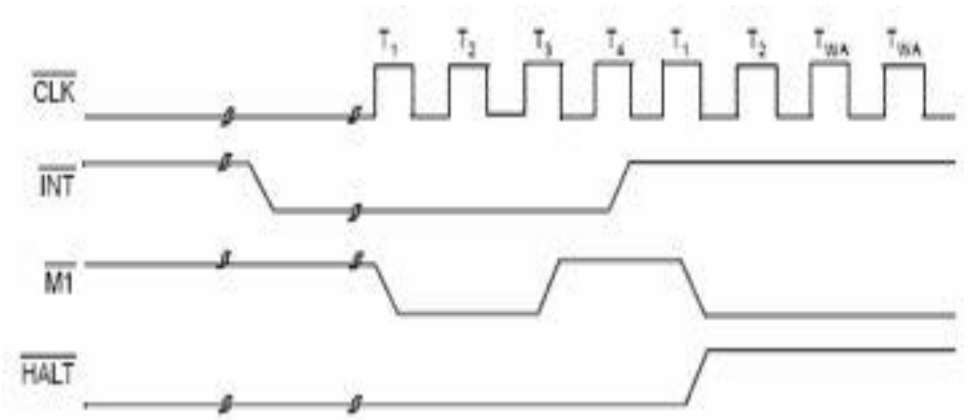
Pada siklus ini sistem clock harus dijalankan terhadap CPU, dengan tujuan melepas (release) state power-down. Jadi ketika sistem clock diberikan pada CPU maka CPU akan melakukan operasi lagi (retart), start dimulai dari kondisi terakhir saat power down. Berikut merupakan diagarm siklus Power-Down Release.



Gambar 3.16. Siklus Power-Down Release 1.



Gambar 3.17. Siklus Power-Down Release 2.



Gambar 3.18. Siklus Power-Down Release 3.

3.14. Respon Interupsi

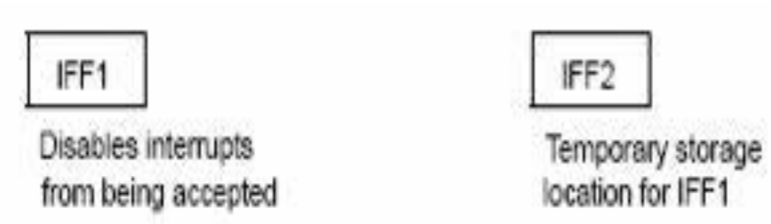
Sebuah interupsi memungkinkan bagi devais periperal untuk menghentikan sementara operasi CPU, dan memaksanya untuk memulai rutin layanan periperal. Rutin layanan ini meliputi pertukaran data, status, kendali informasi antara CPU dengan devais periperal, dan pada saat selesai menjalankan rutin layanan maka CPU kembali menjalankan operasi sebelum terjadi interupsi.

a. Enable/Disable Interupsi

Terdapat 2(dua) buah interupsi pda CPU Z80, yaitu software interupsi maskable (INT) dan interupsi non-maskable (NMI). Untuk interupsi non-

maskable tidak dapat digagalkan (disable) oleh programmer, dan interup ini diterima ketika devais peripheral meminta rutin layanan. Interupsi ini umumnya tersedia untuk layanan fungsi yang sangat penting, dan dapat diset enable atau disable oleh programmer. Dalam CPU Z80 terdapat enable interup flip-flop (IFF) yang dapat di-set atau di-reset oleh programer melalui instruksi

Enable Interrupt (EI) dan Disable Interrupt (DI). Ketika IFF di-reset maka interupsi tidak dapat diterima oleh CPU, kedua enable flip-flop itu adalah IFF1 dan IFF2.



Gambar 3.19. Interupsi Flip-Flop

State IFF1 digunakan untuk inhibit interupsi, sementara itu IFF2 digunakan sebagai tempat penyimpanan tempore untuk IFF1.

Dengan melakukan reset terhadap CPU akan memaksa IFF1 dan IFF2 pada kondisi reset, yaitu interupsi di-disable. Interupsi dapat diset menjadi enable setiap saat melalui instruksi EI dari programer, ketika instruksi EI dijalankan maka semua permintaan interupsi yang ditunda tidak akan diterima sampai setelah instruksi berikut setelah EI dijalankan.

Instruksi tunda diperlukan manakala instruksi berikutnya merupakan instruksi return, dan interup tidak dilayani sampai proses return lengkap. Instruksi EI melakukan setting terhadap kedua flip-flop IFF1 dan IFF2 pada kondisi enable, begitu juga saat CPU menerima interup maskable maka IFF1 dan IFF2 secara otomatis akan reset, dan berikutnya terjadi inhibid terhadap interup sampai programer memberikan instruksi EI.

Tujuan dari IFF2 adalah untuk mengamankan status IFF1 ketika interup non-maskable terjadi, jadi IFF1 akan reset untuk mengjaga interup berikutnya sampai dilakukan re-enable oleh programer. Dengan demikian setelah interup



non-maskable diterima, makainterup maskable disable tetapi state IFF1 sebelumnya telah tersimpan sehingga keseluruhan state CPU memberikan prioritas terhadap interup non-maskable untuk bisa di-restore setiap waktu. Pada saat Register A dimuati data dari Register I (LD A, I) dan instruksi memuati Register A dengan data dari Register R (LD A, R) dijalankan, maka state IFF2 disalin ke parity flag sehingga bisa diuji atau disimpan.

Cara kedua proses restore status IFF1 adalah melalui pelaksanaan Return dari instruksi interup Non-Maskable (RETN). Instruksi ini mengindikasikan bahwa layanan rutin interup non-maskable adalah terselesaikan dan isi IFF2 selanjutnya disalin kembali ke IFF1 sehingga status IFF1 siap menerima interup non-maskable sebagai interup prioritas dan otomatis terjadi restore. Table berikut merupakan rangkuman dampak beberapa instruksi pada kedua buah flip-flop.

Table Interup Enable/Disable, Flip-Flop

Action	IFF1	IFF2	Comments
CPU Reset	0	0	Interup Maskable, INT Disable
Instruksi DI dijalankan	0	0	Maskable INT Disable
Instruksi EI dijalankan	1	1	Maskable, INT Enable
Instruksi LD A,I dijalankan	*	*	IFF2 → Parity Flag
Instruksi LD A,R dijalankan	*	*	IFF2 → Parity Flag
NMI diterima	0	*	Maskable Interup
Instruksi RETN dijalankan	IFF2	*	IFF2 → indikasi layanan rutin interup nonmaskable selesai

b. Respon CPU (Non-Maskable)

CPU selalu menerima dan melayani interup non-maskable, dan ketika interup ini muncul maka CPU akan mengabaikan pelaksanaan instruksi berikutnya dan memulai eksekusi instruksi yang berada pada lokasi 00066H . Pada saat



ini lebih kearah pemanggilan alamat khusus pada memori page 0, sehingga CPU dapat diprogram untuk merespon interup maskable melalui satu dari ketiga mode.

Mode 0

Melalui interupsi mode 0 ini devais dapat menempatkan instruksi pada bus data dan CPU akan mengeksekusinya, sehingga interup yang dilakukan oleh devais akan mendpatkan layanan instruksi berikutnya untuk dilaksanakan. Pada umumnya merupakan instruksi restart karena interupsi devais hanya membutuhkan instruksi dengan byte tunggal, alternatif lain adalah instruksi call 3 byte pada lokasi memori tertentu dilaksanakan.

Jumlah siklus clock untuk melaksanakan instruksi adalah dua lebih dibanding jumlah normal yang digunakan oleh sebuah instruksi, hal ini muncul karena CPU secara otomatis menambahkan dua state wait pada respon siklus interupsi. Keuntungan yang didapat adalah mengatasi ketidak cukupan waktu untuk implementasi rantai eksternal kontrol prioritas. Gambar 9 dan 19 merupakan ilustrasi rinci tetntang timing untuk respon interupsi, paskah aplikasi RESET maka CPU secara otomatis menjalankan interupsi mode 0.

Mode 1

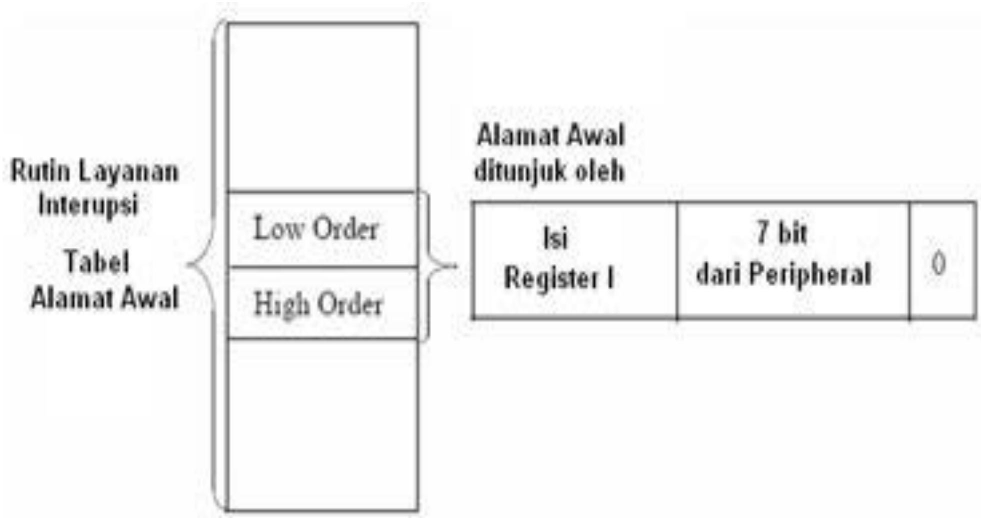
Jika mode 1 dipilih, respon CPU terhadap interup adalah melaksanakan restart pada memori dengan alamat 0038H. Dengan demikian respon yang terjadi identik dengan pelaksanaan interup non-maskable kecuali lokasi aatau alamat yaitu 0038H jadi bukan 0066H. Jumlah siklus yang dibutuhkan untuk menyelesaikan instruksi restart adalah dua lebih dibanding normal dengan tambahan dua state wait.

Mode 2

Mode 2 merupakan mode respon CPU terhadap interup dalam katagori powerful, dengan 8-bit tunggal atau 1 byte dari user melalui indirect call dapat digunakan untuk menunjuk lokasi memori sesuai yang diinginkan (bebas). Dalam mode ini seorang programer dpat mengelolah daftar dari starting addresses 16-bit untuk rutin layanan setiap interup, daftar atau tabel dapat diletakan di lokasi sesuai keinginan programer.

Pada saat interup diterima oleh CPU, maka pointer 16 bit harus diformulasikan untuk mendapatkan alamat awal (starting addresses) dari daftar. Delapan bit teratas merupakan pointer (penunjuk) yang diformulasikan dari isi register I, dan isi register I harus dimuati nilai yang diinginkan oleh programer seperti LD I, A.

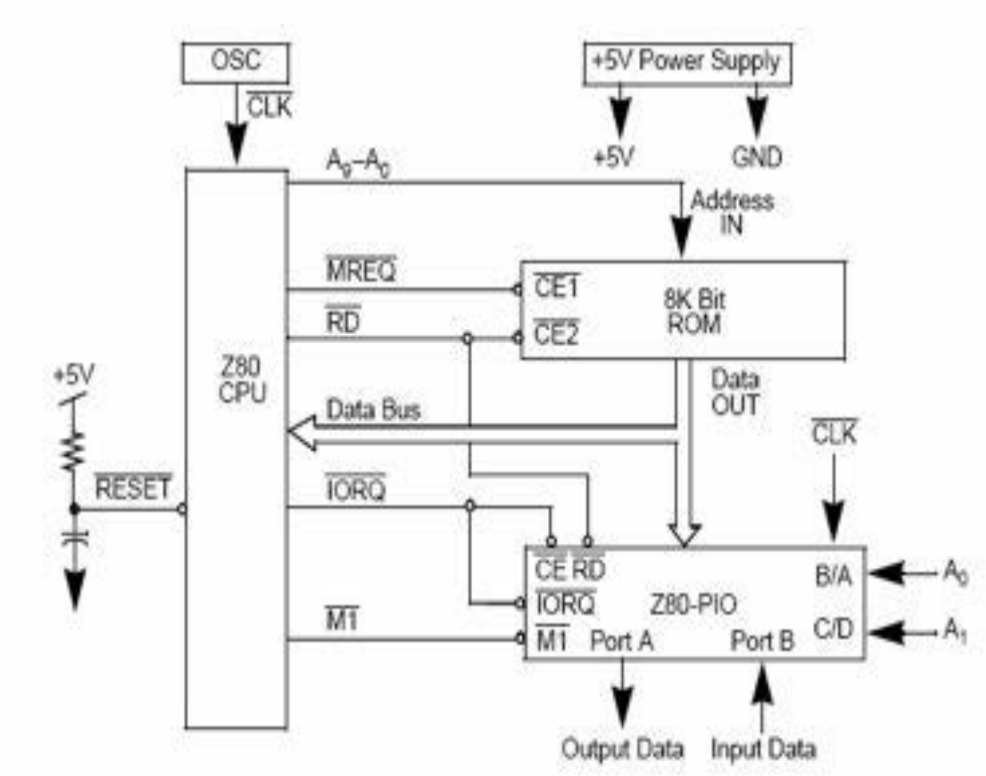
Dalam hal ini CPU melakukan reset untuk proses clear pada register I sehingga nilainya menjadi nol, dan untuk 8 bit terendah merupakan pointer harus diberikan oleh devais yang memberi interupsi. Hanya 7 bit dibutuhkan dari devais penginterup karena LSB harus sama dengan nol, hal ini dibutuhkan karena pointer digunakan untuk mendapatkan dua byte khusus untuk memformulasikan alamat awal rutin layanan interupsi dan alamat awal harus selalui dimulai dari lokasi even.



Gambar 3.20. Mode 2 Rsepon Interupsi

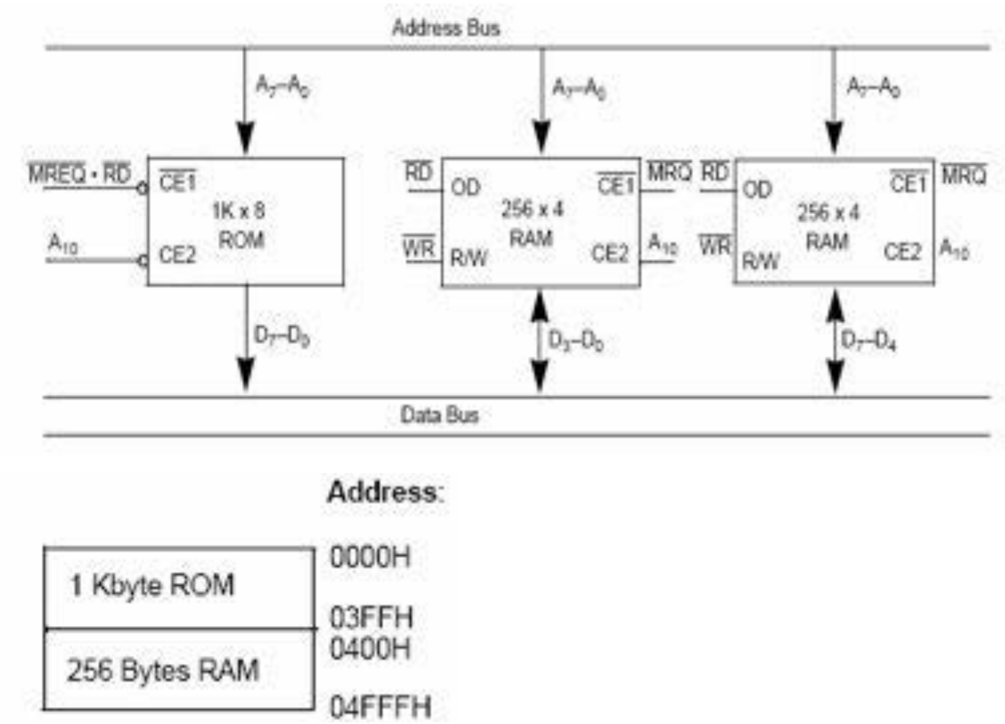
3.15. SISTEM HARDWARE

Berikut merupakan rangkaian minimum yang dibutuhkan oleh Z80 dalam membangun sebuah sistem komputer.



Gambar 3.21. Sistem Minimal Mikroprosessor

Z80 CPU membutuhkan hanya sumber listrik tunggal 5V, oleh karena itu seluruh komponen dalam sistem dapat diaplikasi atau disesuaikan dengan sumber 5V. Untuk eksternal memori dapat dilakukan kombinasi yaitu standar RAM dan ROM atau PROM, pada gambar menunjukan 8 K bit ROM yang berarti sama dengan 1 Kbyte dan blok I/O sehingga memungkinkan CPU berhubungan dengan interface yang berada di luar sistem

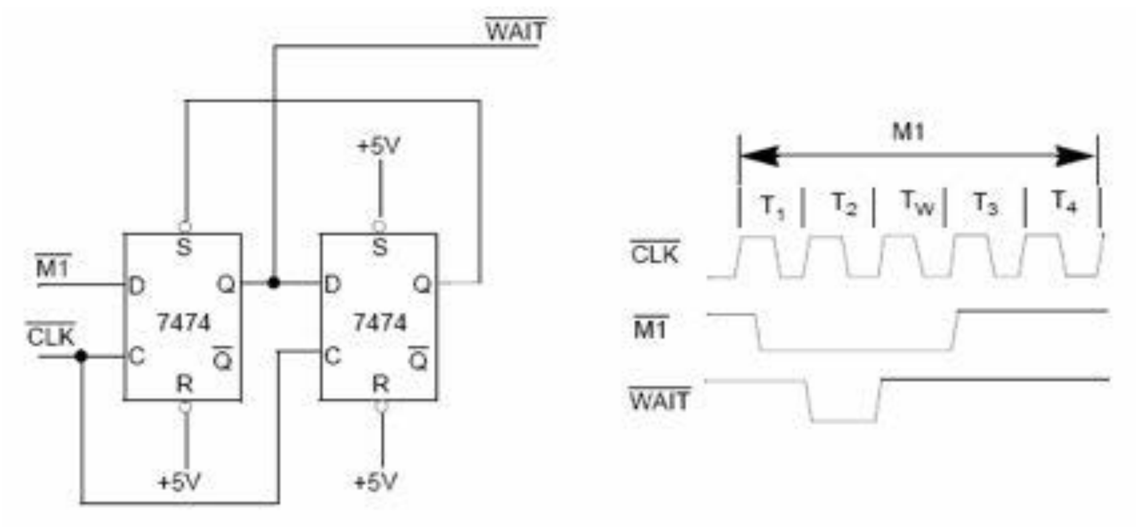


Gambar 3.22. RAM Ekstensi

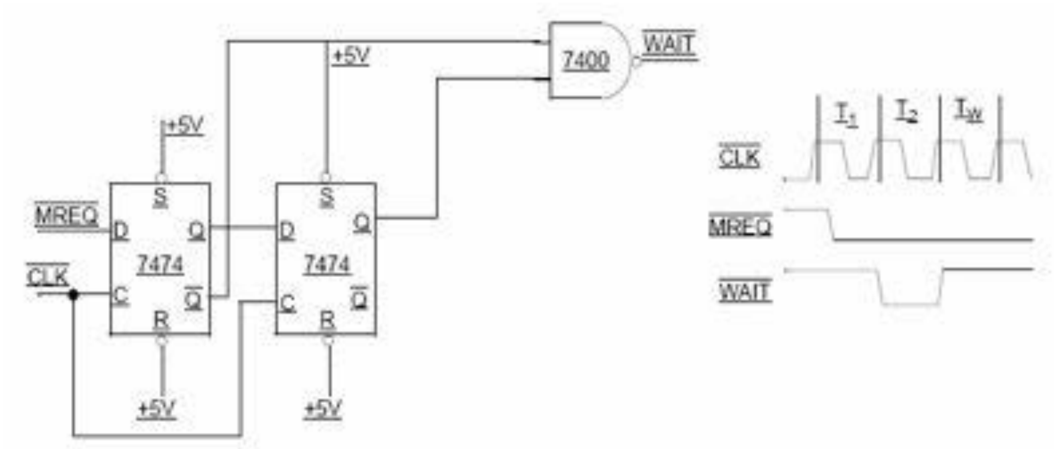
3.16. Kecepatan Akses Memori

Kecepatan akses memori memberikan dampak pada lambat dan cepatnya proses sebuah program aplikasi, jalur WAIT pada CPU memungkinkan operasi Z80 dengan berbagai kecepatan akses memori. Oleh karena itu dibutuhkan waktu untuk akses memori dalam melakukan layanan operasional siklus instruksi

M1, siklus akses memori secara komplit adalah satu setengan siklus clock. Rangkaian berikut merupakan rangkaian sederhana sesuai dengan tujuan yang diharapkan, rangkaian bisa diubah dengan memasukan state wait tunggal pada setiap akses memori seperti ditunjukan gambar berikut:



Gambar 3.23. Siklus Akses Memori.

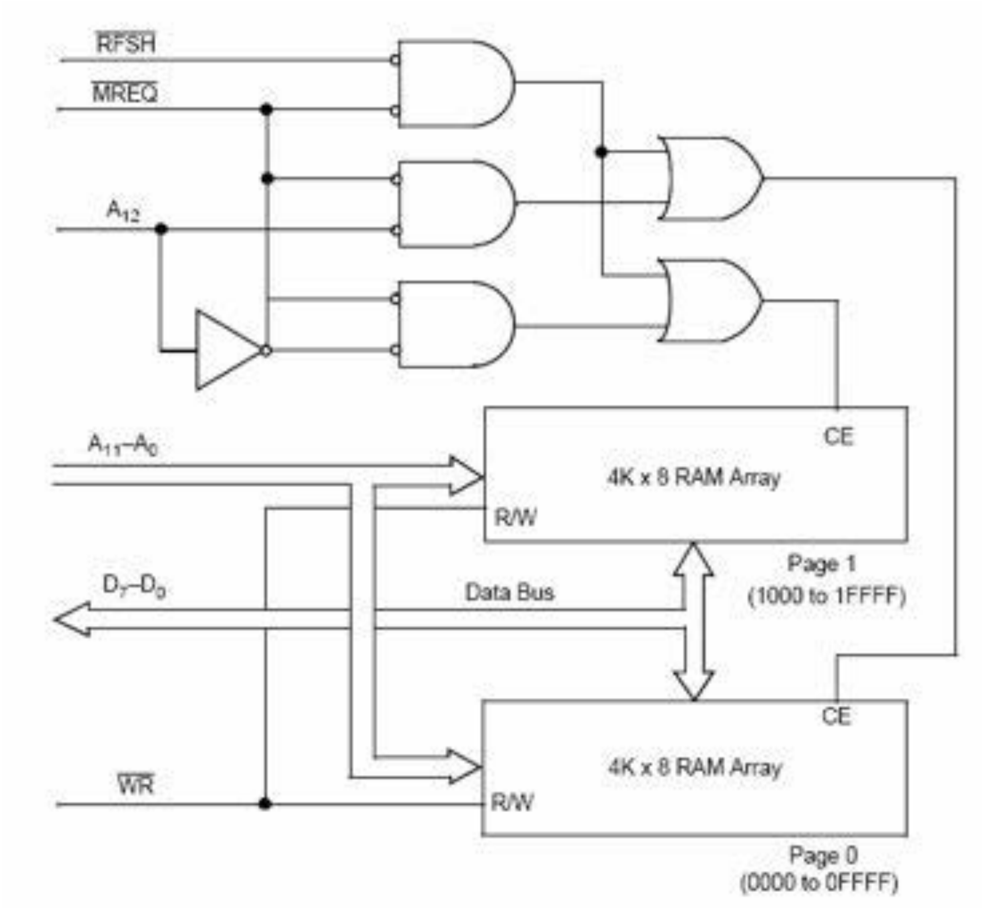


Gambar 3.24. Penambahan satu state Wait State pada siklus memori

3.17. INTERFACE MEMORI DINAMIK

Setiap RAM dinamik memiliki spesifikasi masing-masing, artinya antara RAM yang satu berbeda dengan RAM lainya. Gambar 21 merupakan ilustrasi rangkaian logika interface untuk 8Kbyte RAM dinamik menggunakan 18-pin memori 4Kbyte, rangkaian ini RAM dianggap hanya memori dari sistem sehingga A12 digunakan untuk memilih diantara dua halaman memori. Dalam waktu refresh semua memori dalam sistem harus dibaca, CPU memberikan

layanan refresh secara benar alamat pada jalur A0 sampai A6. Jika penambahan memori lebih besar pada sistem maka harus diganti dua gerbang yang beroperasi pada A12, yaitu dengan dekoder untuk memenuhi semua bit alamat. Bufer untuk bus alamat dan bus data dibutuhkan manakala diperlukan untuk sistem yang besar.



Gambar 3.25. Interface Memori Dinamik

3.18. PENGALAMATAN MEMORI

Sebuah mikroprosesor tidak dapat berdiri sendiri dalam operasinya, artinya mikroprosesor tersebut membutuhkan tambahan komponen berupa memori yang fungsinya untuk menyimpan data dan program yang akan dijalankannya.

Berbicara memori maka tidak bias lepas dengan sistem pengalamatan memori, dengan mengetahui hal tersebut maka dapat ditentukan seberapa besar kapasitas pengalamatan memori. Secara fisik memori diproduksi dalam 2(dua) katagori yaitu RAM dan ROM, dan terdapat banyak tipe yang dapat ditemui di pasaran. Dalam sistem mikroprosesor Z80 banyak dikembangkan dengan tipe memori RAM 6116 dan EPROM 2716, untuk itu perlu bagi pengembang sistem mikroporosesor dapat menentukan dan menjelaskan fungsi masing - masing pin dari pada RAM 6116 dan EPROM 2716, menjelaskan cara pengalamatan RAM 6116 dalam operasi dasarnya, menjelaskan proses operasi dasar RAM 6116, menjelaskan cara pengalamatan EPROM 2716 dalam operasi dasarnya, menjelaskan proses operasi dasar EPROM 2716, dan merencanakan pendekode pengalamatan RAM maupun EPROM.

1. Kapasitas Pengalamatan memori

Kapasitas penyimpan pada RAM atau EPROM tergantung pada jumlah PIN alamat (Ao - An) dari RAM/EPROM tersebut, dan dihitung dengan rumus :

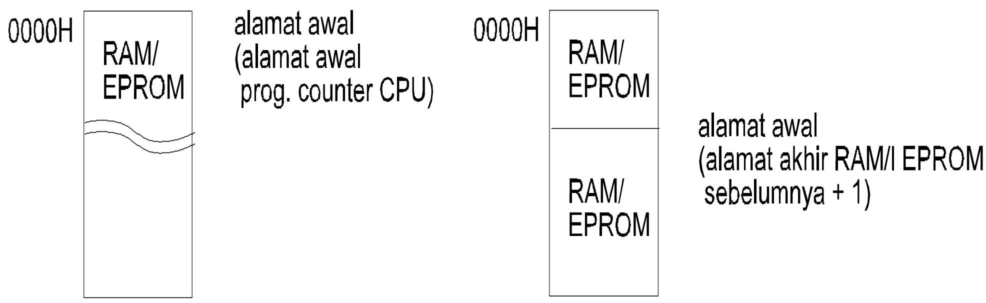
Kapasitas Penyimpan = 2^{n+1}

Sebagai contoh :

* Jumlah pin sebuah RAM 6116 = sebanyak 11 buah (Ao ~ A_{10})
* Maka kapasitas RAM ini adalah :

$2^{(10+1)} = 2^{11} = 2048$ lokasi

Untuk menentukan alamat awal dan akhir dari penyimpan atau memori di atas dapat ditentukan sebagai berikut :



Gambar 3.26. Penentuan alamat awal dan akhir memori

Gambar 3.26 merupakan gambaran penentuan alamat awal dan akhir memori, yaitu dilakukan dengan cara sebagai berikut:

- Alamat awal dapat ditentukan 0000 H yaitu alamat awal program counter dari CPU atau alamat akhir RAM atau EPROM sebelumnya ditambah 1.
- Alamat akhir dapat ditentukan sesuai dengan jumlah kapasitas RAM/EPROM tersebut ditambah dengan alamat awalnya.

Contoh 1 :

Mencari alamat akhir dari pengalamatan RAM atau EPROM yang berkapasitas 1 K Byte (1024 Byte), bila alamat awal pengalamatan = 10 buah (Ao ~ A9)

A9	**A8**	**A7**	**A6**	**A5**	**A4**	**A3**	**A2**	**A1**	**A0**	**Alamat**
512	**256**	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**	
0	0	0	0	0	0	0	0	0	0	000 H = alamat awal
1	1	1	1	1	1	1	1	1	1	+ 3FFH = kapasitas maks
1	1	1	1	1	1	1	1	1	1	3FFH = alamat akhir

Kapasitas : 512 + 256 + 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 + 1 = 1024 lokasi

(alamat 000 dihitung sebagai 1 lokasi)

Contoh 2 :

Mencari alamat akhir dari pengalamatan RAM atau EPROM yang berkapasitas 2 K Byte, bila alamat awal pengalamatan = 400 H

Jumlah PIN alamat = 11 buah (Ao - A10)

A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Alamat
2048	**1024**	**512**	**256**	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**	
0	1	0	0	0	0	0	0	0	0	0	0	400 H = alamat awal
0	1	1	1	1	1	1	1	1	1	1	1	+ 7FFH= kapasitas maks
1	0	1	1	1	1	1	1	1	1	1	1	BFFH = alamat akhir

Kapasitas =1024 + 512 + 256 + 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 + 1

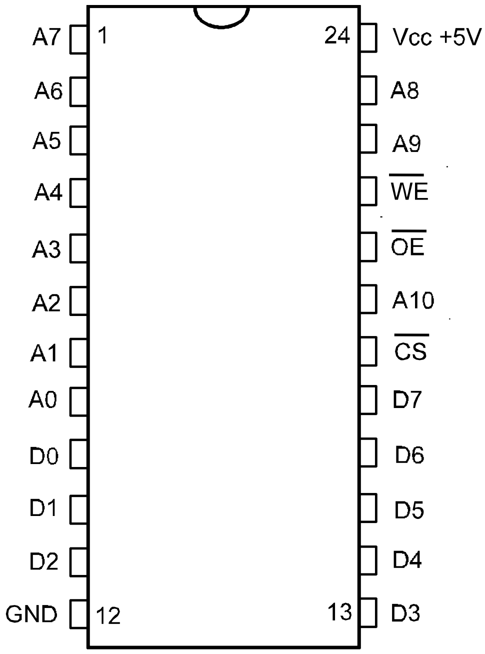
=2048lokasi

2. Konfigurasi Pin IC Memori

Setiap IC atau chip yang digunakan dalam teknik elektronika maupun teknik digital serta teknik mikroprosesor memiliki pin kaki yang berbeda, dan perbedaan tersebut dibuat berdasarkan kepentingan dan kegunaan fungsi IC.

Oleh karena dibutuhkan chip IC dengan tipe RAM 6116 maka perlu dilihat konfigurasi pin kaki IC, sehingga dalam melakukan penyambungan dengan komponen yang lain dalam membentuk sistem dapat dilakukan dengan baik dan benar. Baik dalam arti estetika rangkaian termasuk didalamnya ketraturan rangkaian, sedangkan benar berarti rangkaian harus sesuai (cocok) dengan fungsinya karena jika salah maka rangkaian tidak akan berfungsi.

Gambar x berikut menunjukan konfigurasi fungsi pin pada memori statis tipe RAM 6116:



Keterangan pin :

VCC = + 5 Volt

$\overline{CS}$ = Chip Select (aktif low)

$\overline{OE}$ = Output Enable (aktif low)

$\overline{WE}$ = Write Enable (aktif low)

GND = Ground

A0 ~ A10 = Masukan alamat

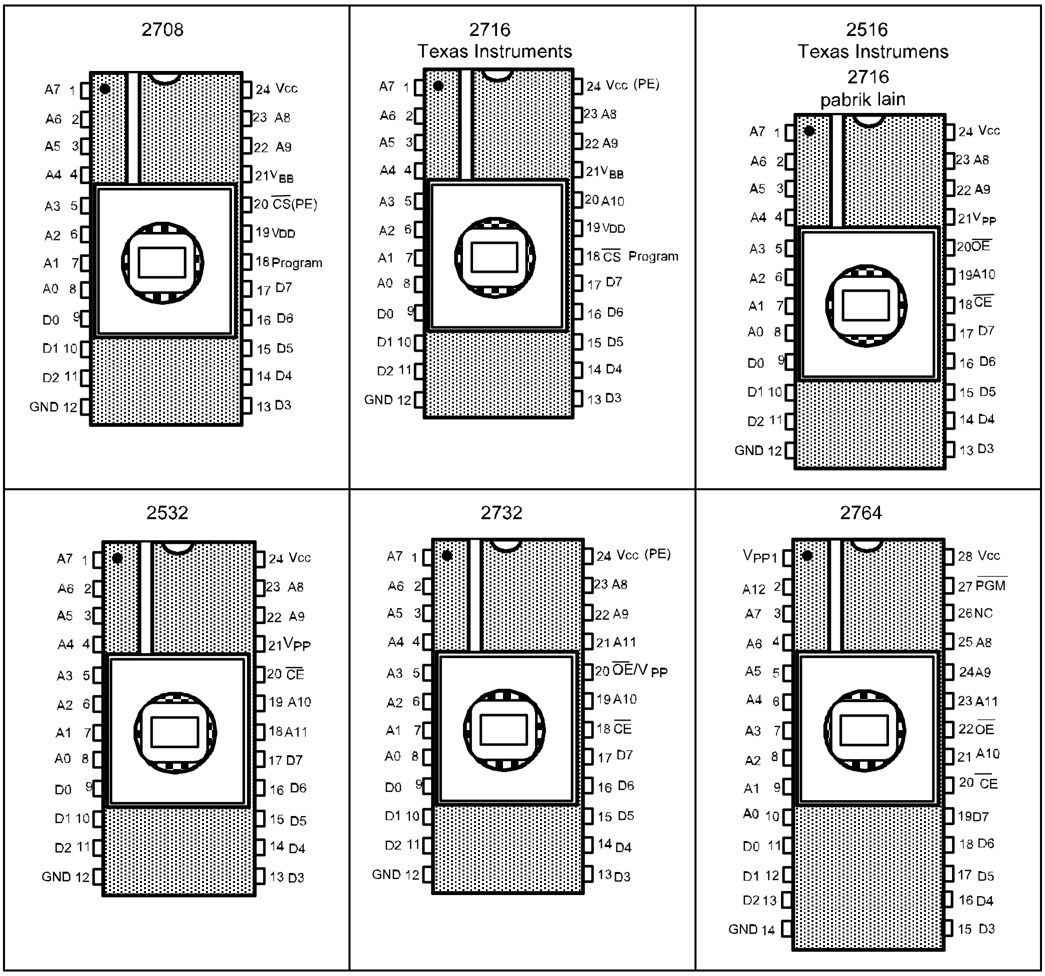
D0 ~ D7 = Input output data

Gambar 3.27. Konfigurasi pin RAM 6116

Pin kaki nomor 1 sampai dengan nomor 8 ditambah nomor 19, 22 dan 23 berfungsi untuk menunjuk alamat atau lokasi data yang tersimpan dalam IC tersebut, berdasarkan kombinasi pin kaki tersebut alamat lokasi penyimpanan adalah 000 sampai 3FF.

$\overline{WE}$ digunakan dengan sinyal low saat menuliskan data ke dalam memori (write), $\overline{CS}$ digunakan dengan sinyal low saat chip akan ditulisi data atau dibaca datanya, $\overline{OE}$ digunakan dengan sinyal low saat chip akan dibaca datanya. Data yang dituliskan ke dalam memori dan data yang dibaca dari memori disalurkan melalui D0 ~ D7.

Gambar x berikut menunjukan beberapa konfigurasi pin kaki IC dengan tipe EPROM 2708, 2716, 2516, 2532, 2732 dan 2764. Tipe IC tersebut mengindikasikan kapasitas lokasi memori yang digunakan untuk penyimpanan data, hal ini bias dilihat atau ditinjau dari jumlah pin yang berlabel A yang artinya address (alamat).



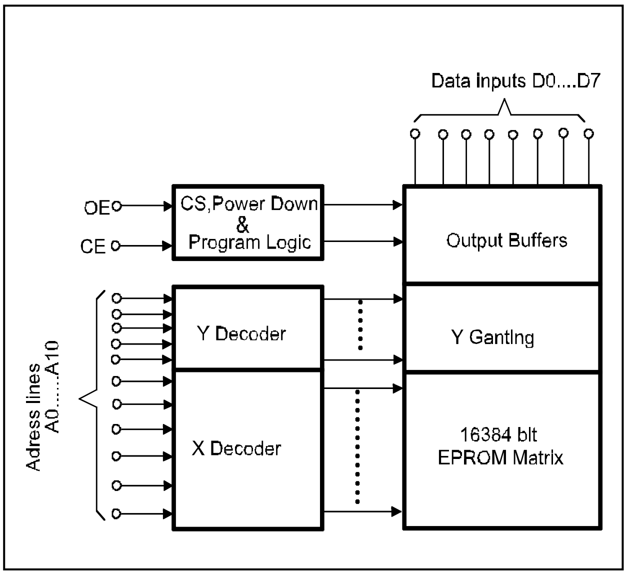
Gambar 3.28. Konfigurasi pin EPROM

Ketrangan Gambar:

Kapasitas Memori	Susunan pin
2708 1024 x 8 bit.	Vcc = + 5 V
2716 2048 x 8 bit.	VBB = - 5 V
2732 4096 x 8 bit.	VDD = + 12 V
2764 8192 x 8 bit.	Vpp = + 5 V dalam ragam siap
2516 2048 x 8 bit.	+ 25 V dalam ragam pengacaraan

2532 4096 x 8 bit .	A0... An = Jalan masuk alamat D0... D7 = Masuk dan keluar data CS = Chip Select CE = Chip Enable OE = Output Enable PD = Power Down

Diagram Blok sebuah 2716



Secara diagram blok rangkai-an internal EPROM 2716 termasuk familinya seperti ditunjukan pada gambar x, dimana saluran pemilih alamat A0-A10 dihubungkan rangkai-an decoder internal yaitu A0 sampai A5 dengan decoder X dan A6 sampai A10 dengan decoder Y.

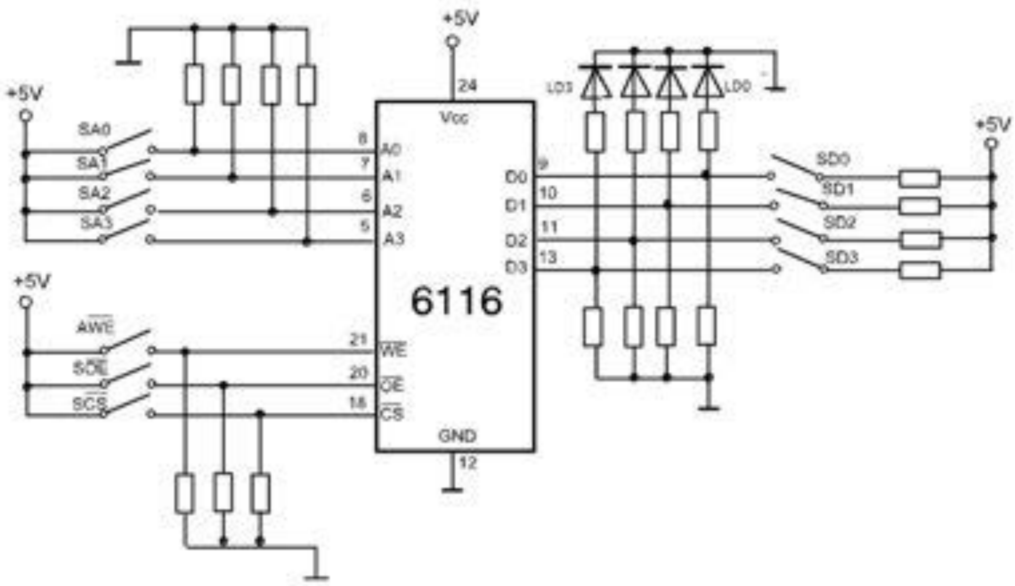
Gambar 3.29. Diagram blok EPROM 2716.

Kedua decoder X dan decoder Y berfungsi untuk menunjuk lokasi tempat penyimpanan data dalam memori dengan metode matrik.

Saluran OE dan CE berfungsi untuk mengendalikan chip IC saat dipilih untuk baca dan dan kendali dilakukan terhadap buffer data.

3. Pengalamatan RAM 6116 dalam operasi dasar

Operasi dasar yang dilaksanakan pada RAM : adalah operasi penulisan data atau pembacaan data ke / dari RAM oleh CPU. Data yang tersimpan sifatnya sementara, tergantung pada catu daya pada RAM



Gambar 3.30. Model operasi RAM 6116

Gambar 3.30 menunjukan rangkaian input dan output pada RAM 6116 yang dihubungkan dengan saklar sebagai input dan LED sebagai output, proses pembacaan data dan proses penulisan data dilakukan pada saluran yang sama yaitu D0 sampai D3 (4 bit).

Penunjuk lokasi dilakukan melalui saluran input alamat yaitu A0 sampai A3, yang berarti lokasi yang bisa diakses baik saat pembacaan maupun penulisan data adalah alamat 0 sampai 16.

Untuk saluran kontrol penulisan atau pembacaan dilakukan melalui saluran baca dan tulis serta penunjuk aktif chip (chip select).

Proses jalannya operasi dasar RAM 6116

a. Proses Penulisan Data.



1. Tentukan data pada Bus Data (SD3 - SD0)

Contoh : 6 H

2. Tentukan Alamat Penyimpan (SA3 - SA0)

Contoh : OH

3. Saklar $\overline{SWE}$ dibuka $\rightarrow$ operasi menulis
4. Saklar $\overline{SOE}$ ditutup $\rightarrow$ operasi output ditutup
5. Saklar $\overline{SCS}$ ditutup $\rightarrow$ operasi IC dipilih
6. Ulangi proses penulisan diatas (langkah 1 - 5) untuk mengisi alamat lainnya yaitu 4 H dengan data EH (catu jangan diputuskan pada proses ini)

b. Proses Pembacaan Data

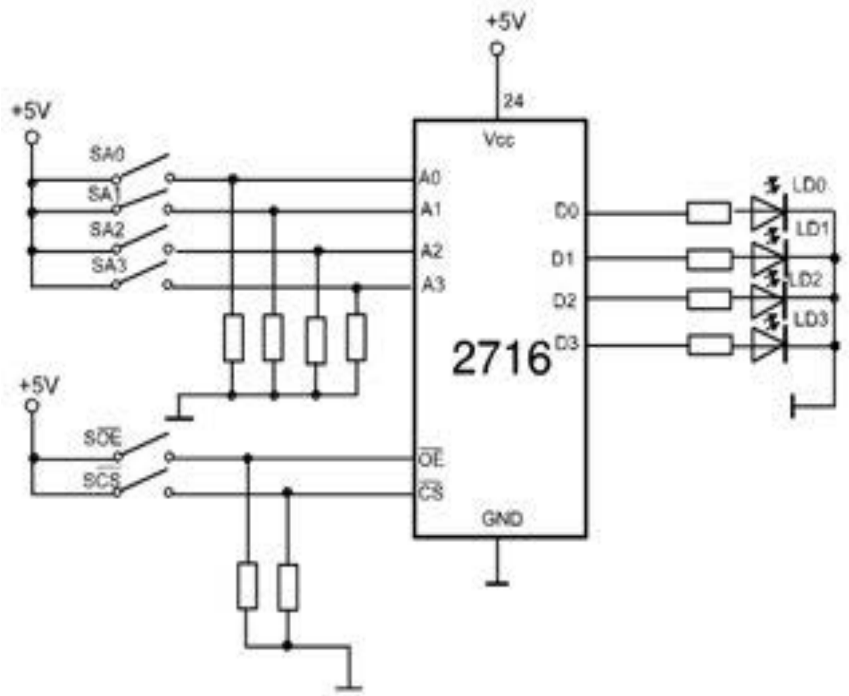
1. Posisi sakelar SD3 - SD0 pada posisi terbuka semua
2. Tentukan Alamat Penyimpan (SA3 - SA0) yang akan dibaca datanya

Contoh : OH

3. $\overline{SWE}$ $\rightarrow$ ditutup
4. $\overline{SOE}$ $\rightarrow$ di buka $\rightarrow$operasi membaca
5. $\overline{SCS}$ $\rightarrow$ ditutup - dibuka
6. Pada LED LD3 - LD0 akan menunjukkan data 6 H
7. Ulangi proses pembacaan diatas (langkah 1 - 6) untuk membaca isi alamat penyimpan lainnya. Yaitu : 4 H.

Data yang akan ditunjukkan pada LED LD3 - LD0 adalah EH.

4. Pengalamatan EPROM 2716 dalam operasi dasar



Gambar 3.31. Rangkaian uji pembacaan data EPROM 2716

Operasi dasar yang dapat dilakukan pada EPROM adalah hanya operasi pembacaan data dari EPROM oleh CPU.

Data tersimpan tetap paten pada EPROM dan tidak tergantung pada catu daya . Pengisian data pada EPROM dilakukan dengan mempergunakan EPROM Writer/Programer.

Jalannya Operasi Dasar (Proses Pembacaan) EPROM 2716

1. Tentukan alamat penyimpan (SA3 - SA0) yang isinya akan dibaca
2. $\overline{\text{SOE}}$ $\rightarrow$ di buka
3. $\overline{\text{SCS}}$ $\rightarrow$ ditutup - dibuka
4. Pada LD3 - LD0 akan menunjukkan isi alamat yang dipilih
5. Ulangi langkah 1 - 4 untuk membaca data pada alamat lain
6. Putuskan catu daya chip 2716
7. Ulangi langkah 1 - 5 untuk membaca data alamat - alamat yang sama



pada langkah 1 - 5

8. Hasil pada LD3 - LD0 menunjukkan data yang sama walaupun catu daya telah diputuskan.

5. Pengalamatan RAM 6116 dan EPROM 2716 pada Sistem Minimal Z - 80

Dalam pengalamatan ini , beberapa pin masukan dari CPU Z - 80, juga dipergunakan dalam pengalamatan RAM dan EPROM ini.

Selain pin - pin kontrol $\overline{WR}/\overline{WE}$, $\overline{RD}/\overline{OE}$ dan Bus Data, dari CPU, juga digunakan pin - pin alamat

- A15 - A0 dan $\overline{MREQ}$
- A15 - A0 dipergunakan untuk memberikan data alamat RAM/EPROM.
- $\overline{MREQ}$ digunakan bersama sinyal - sinyal alamat A15 - A0 untuk mengaktifkan RAM/EPROM.

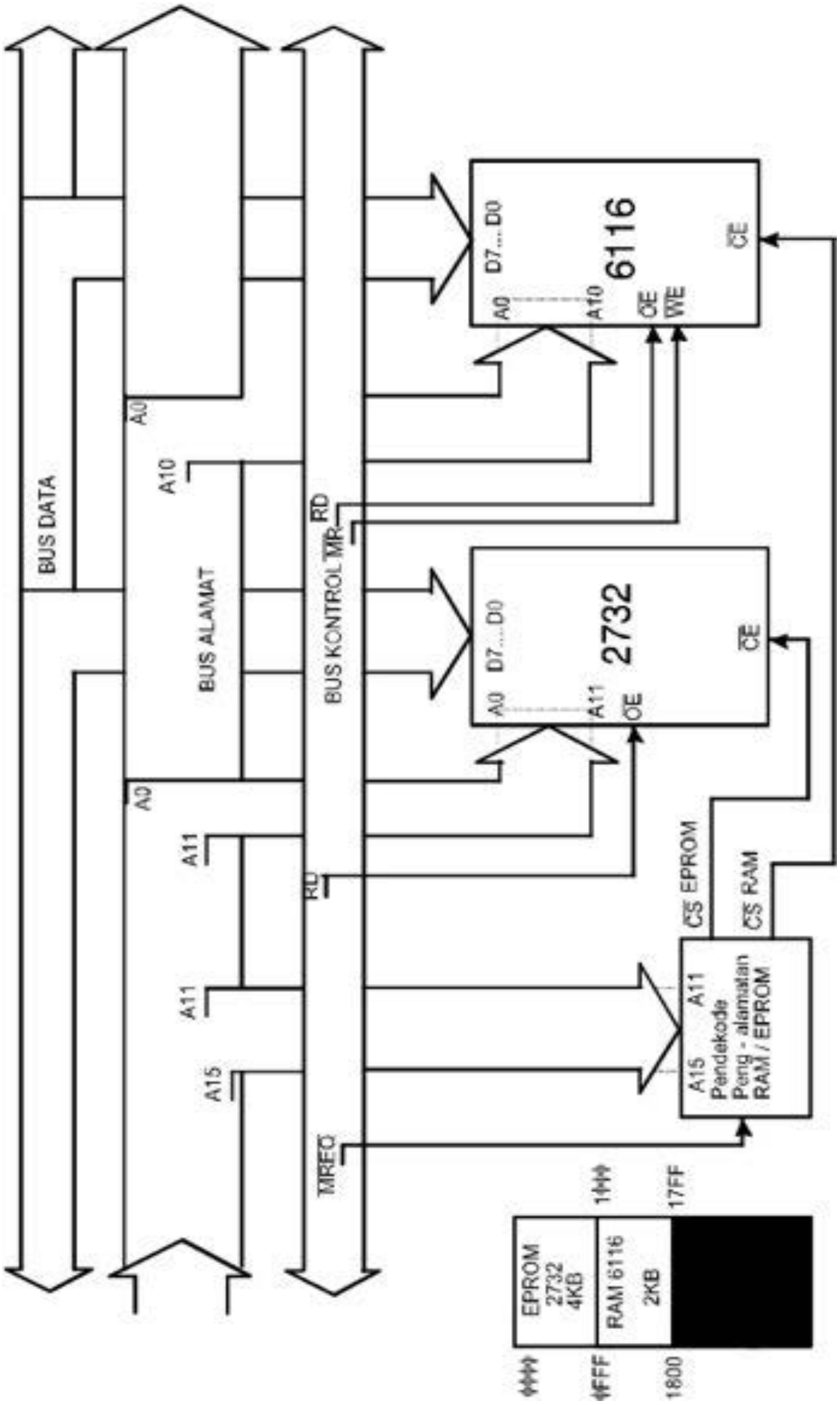
Rangkaian pendekode pengalamatan RAM/EPROM berfungsi untuk mengaktifkan RAM/EPROM pada daerah pengalamatannya, yaitu mulai dari alamat awal sampai alamat akhir dari RAM/EPROM, sesuai dengan peta pengalamatannya.

Pin - pin alamat CPU yang tidak termasuk dalam daerah pengalamatan RAM/EPROM, harus diperhatikan dan diikutkan dalam pengalamatan RAM/EPROM.

Untuk menghindari adanya beberapa alamat RAM atau EPROM yang menunjuk pada data lokasi RAM/EPROM yang sama, oleh sebab itu pin - pin alamat CPU ini bersama - sama dengan sinyal

$\overline{MREQ}$ dipergunakan sebagai masukan dari pendekode pengalamatan RAM/EPROM.

Hasil pendekodean alamat (keluaran pendekode pengalamatan), dihubungkan ke pemilih Chip ($\overline{CS}$ / CE) dari masing - masing Chip.


BUS DATA
BUS ALAMAT
BUS KONTROL
6116
2732
D7....D0
A0
A10
A11
A15
RD
MR
MREQ
OE
WE
CE
CS EPROM
CS RAM
A15 A11
Pendekode
Peng - alamatan
RAM / EPROM
EPROM
2732
4KB
RAM 6116
2KB
17FF
1800

Gambar 3.32 Rangkaian pengalamatan RAM/EPROM

6. Perencanaan Pendekode Pengalamatan RAM/EPROM

a. Pemetaan Lokasi Pengalamatan

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Daerah Memori
32.768	16.384	8.192	4.096	2.048	1.024	512	256	128	64	32	16	8	4	2	1	(Alamat)
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000 H awal EPROM
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0FFFH akhir EPROM
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1000H awal RAM
0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	17FFH akhir RAM

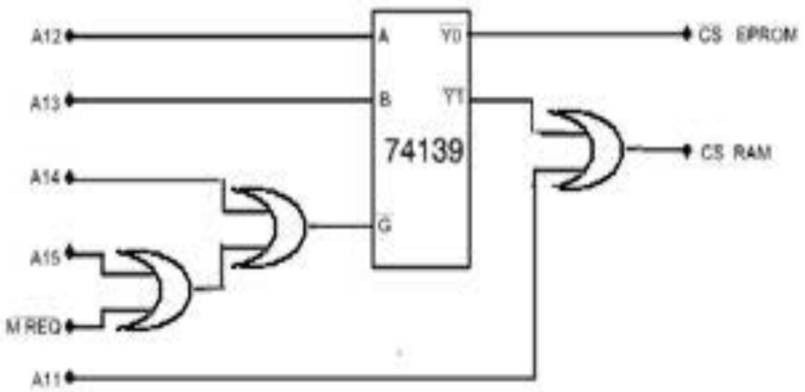
b. Persamaan Boole

$$\overline{CS}\,\text{EPROM} = ((\overline{MREQ} \vee \overline{A15}) \vee \overline{A14} \vee \overline{A13} \vee \overline{A12})$$

$$\overline{CS}\,\text{RAM} = (((\overline{MREQ} \vee \overline{A15}) \vee \overline{A14}) \vee \overline{A13} \vee A12) \vee \overline{A11})$$

→ Bekerja dengan negatif → yang dicari

c. Rangkaian Pendekode Pengalamatan RAM /EPROM



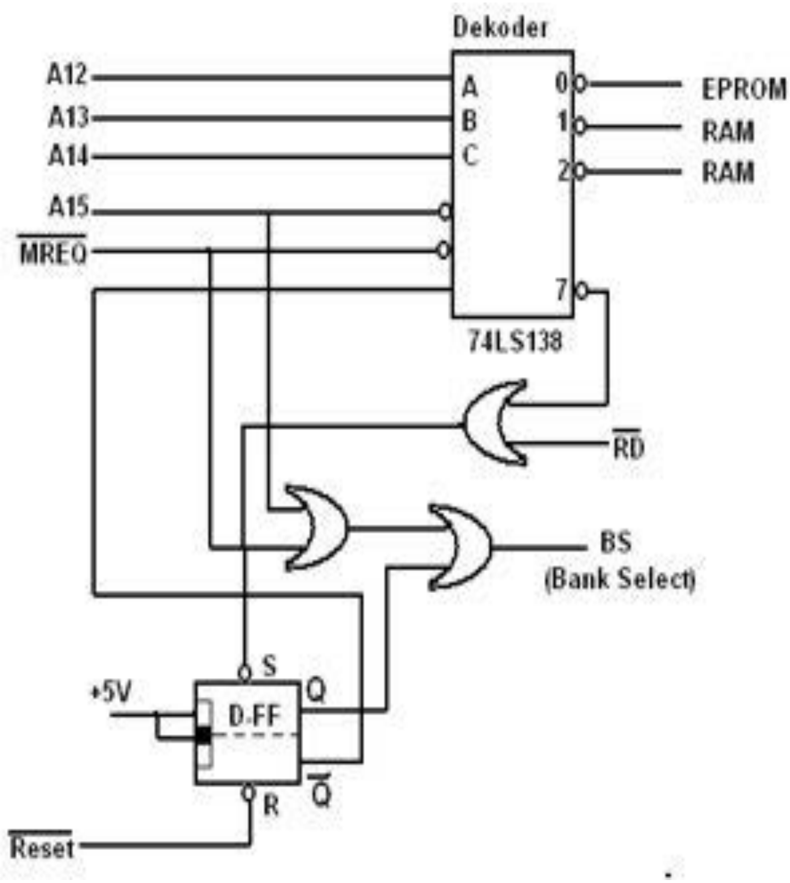
Gambar 3.33. Rangkaian dekoder pengalamatan RAM/EPROM

7. **Pengalamatan Bank Memori**

Dalam mikroposesor dapat menunjuk alamat sampai dengan 64 K alamat memori, yaitu melalui 16 bit jalur penunjuk alamat. Sehingga untuk penunjukan alamat lebih dari 64 K diperlukan rangkaian pengalamatan secara khusus, yaitu dengan memanfaatkan metode sistem bank memori.

Sebagai alamat awal memori yaitu dengan alamat 0 (nol) dan pada alamat inilah suatu program monitor diletakan, oleh karena itu diperlukan sebuah ROM atau EPROM dan pada alamat inilah pertama kali dijalankan manakal tombol reset ditekan.

Sedangkan RAM diperlukan oleh sistem mikroprosesor untuk menuliskan program ataupun data, dengan demikian program awal yang diletakan pada lokasi EPROM kerjanya akan dilanjutkan oleh program yang dituliskan pada RAM.



Gambar 3.34. Rangkaian dekoder bank memori

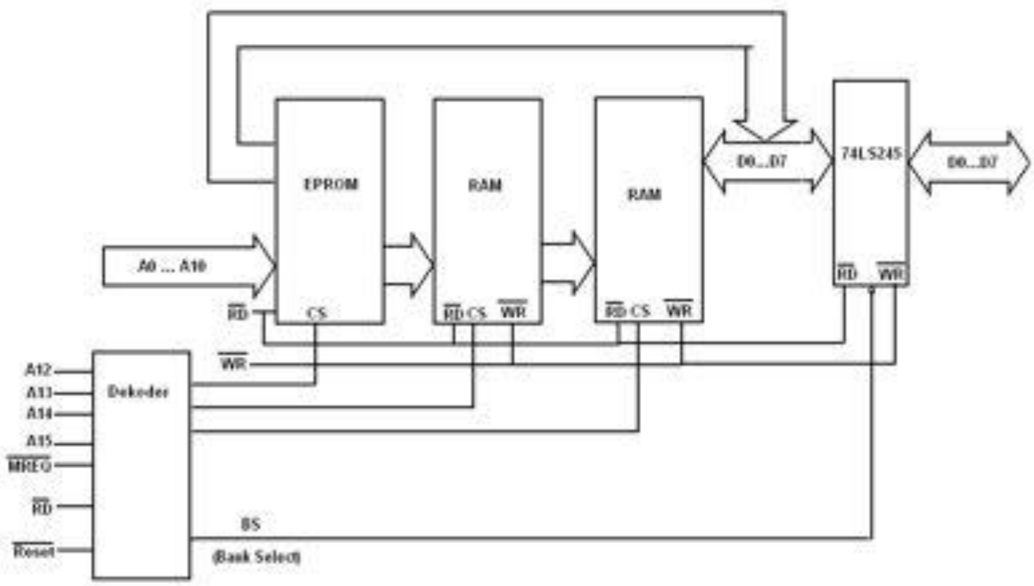
Gambar 3.34 merupakan rangkaian dekoder bank memori, pada sistem bank memori jalur alamat (*address bus*) yang terdiri dari 16 bit jalur dibagi menjadi dua.

Untuk 11 jalur digunakan langsung menunjuk alamat pada chip IC RAM atau EPROM dan 4 bit berikutnya digunakan untuk memilih *chip* IC mana yang diaktifkan.

Untuk pemilihan nomor bank memori yang ditunjuk digunakan A15 dan $\overline{MREQ}$, seperti ditunjukan oleh gambar x rangkaian penunjuk *chip* IC baik RAM maupun EPROM dikendalikan oleh A12, A13 dan A14.

Penunjukan bank memori dihasilkan dari logika kombinasi A15 dan $\overline{MREQ}$ yang dikombinasikan dengan keluaran Q dari D-FF, untuk men-set D-FF dilakukan melalui Y7 dari dekoder 74LS138 yang dikombinasikan dengan sinyal baca yaitu logika dari jalur kontrol $\overline{RD}$.

Dengan demikian jika diinginkan sebua sistem pengalamatan memori untuk EPROM dengan alamat 0000 sampai 0FFF, kemudian untuk memori RAM dengan alamat 1000 sampai alamat 1FFF, dan satu buah memori RAM lainnya dengan alamat 2000 sampai alamat 23FF maka rangkaian pengalamat tersebut di atas dapat digunakan dan secara blok ditunjukan pada gambar 3.35.



Gambar 3.35. Pengalamatan sistem bank memori

Latihan

1. Jika jumlah pin sebuah RAM (Misal 6116) = sebanyak 10 buah (A_0 - A_9)

   Berapa kapasitas RAM tersebut ?

2. Bagaimana cara mengetahui alamat akhir dari RAM tersebut (soal no.1) jika alamat awal RAM 6116 adalah 000H ?

3. Berapakah alamat akhir dari RAM jika, alamat RAM 6116 adalah 200H ?

Jawaban

1. Rumus kapasitas penyimpan 2^{n+1}

   $2^{(9+1)} = 2^{10} = 1024$ lokasi

2. Mencari alamat akhir dari RAM tersebut

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Alamat
512	**256**	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**	
0	0	0	0	0	0	0	0	0	0	000 H = alamat awal
1	1	1	1	1	1	1	1	1	1	+ 3FFH = kapasitas maks.
1	1	1	1	1	1	1	1	1	1	3FFH = alamat akhir

Kapasitas : 512 + 256 + 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 + 1 = 1024 lokasi (1 k Byte)

3. Jika alamat awal 200 h

A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Alamat
1024	512	256	128	64	32	16	8	4	2	1	
0	1	0	0	0	0	0	0	0	0	0	200 H = alamat awal
0	1	1	1	1	1	1	1	1	1	1	+3FFH = kapasitas maks
1	0	1	1	1	1	1	1	1	1	1	5FFH = alamat akhir

Kapasitas : 512 + 256 + 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 + 1 = 1024 lokasi (1 K Byte)

3.19. Programmable Pheriperal Interface (PPI) 8255.

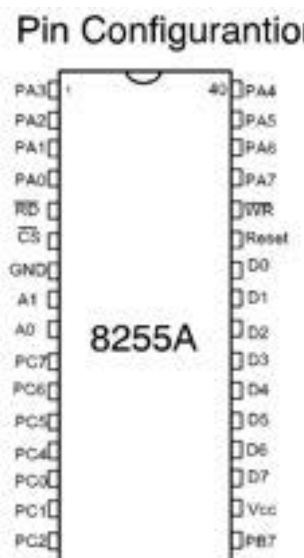
Untuk dapat memahami prinsip kerja perantara pheriperal terprogram 8255 dapat dilakukan pendekatan teoritis dan praktis, yang meliputi fungsi PPI 8255, sistem pengontrolan port, pengoperasian PPI 8255, fungsi kontrol interrupt, sistem mode operasi yang digunakan, sistem sinyal kontrol masukan, sistem sinyal kontrol keluaran.

1. Fungsi Pin PPI 8255

PPI 8255 terdiri dari 4 register port yang menampung data 8 bit dan berhubungan dengan bus data sistem melalui bus data internal. Dalam register ini ditempatkan data masukan, keluaran atau data kata kendali.

Masing-masing register mempunyai alamat sendiri yang dapat dipilih melalui pengkodean pengalamatan PPI 8255.

Gambar 3.36 berikut menunjukkan Pin - Pin dari PPI 8255 dengan fungsinya masing – masing



Pin Names

D7 - D0	Data Bus (Bi-Directional)
RESET	Reset Input
$\overline{CS}$	Chip Select

$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A0, A1	Port Adress
PA7-PA0	Port A (BIT)
PB7-PB0	Port B (BIT)
PC7-PC0	Port C (BIT)
Vcc	+5 Volts
GND	0 Volt

Gambar 3.36. Konfigurasi pin PPI 8255

Deskripsi fungsi pin PPI 8255

Pin - Pin Saluran Data :

- Bus Data : D7 - D0
- Bus Port A : PA7 - PA0
- Bus Port B : PB7 - PB0
- Bus Port C : PC7 - PC0

Pin - Pin Saluran Pengontrol :

- Baca (Read) : $\overline{RD}$
- Tulis (Write) : $\overline{WR}$
- Reset : Reset

Pin-pin pendekode alamat :

Pin - pin yang sangat penting untuk mengkode alamat register PPI 8255 adalah :

Pin alamat : A1 dan A0 serta pemilih chip (Chip select) : CS

keterangan fungsi masing-masing pin dan penggunaannya dalam rangkaian

D7 - D0 : Dihubungkan ke sakelar dan LED



Sakelar dan LED menggantikan fungsi Bus Data sebagai jalannya data 2 arah (membaca dan menulis).

Untuk rangkaian ini pada saat operasi membaca data, posisi sakelar SD7 - SD0 harus terbuka.

PA7 - PA0 : Dihubungkan ke LED

Konfigurasi PPI 8255 mengatur port A sebagai terminal keluaran data dan LED dipakai untuk menampilkan data keluaran pada terminal port A.

PB7 - PB0 : Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan data ke terminal port B.

$\overline{RD}$: Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan sinyal baca pada terminal $\overline{RD}$

$\overline{WR}$: Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan sinyal tulis pada terminal $\overline{WR}$

Sakelar Write (S $\overline{WR}$) terbuka : operasi menulis

RESET : Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan sinyal Reset pada terminal Reset

Sakelar Reset (S RS) terbuka : PPI tidak terreset

$\overline{CS}$: Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan sinyal Pemilihan Chip pada terminal ($\overline{CS}$) Chip Select

Sakelar $\overline{CS}$ terbuka : PPI aktif

A1-A0 : Dihubungkan ke sakelar

Sakelar dipakai untuk memasukkan data alamat Port

2. Operasi Dasar PPI 8255

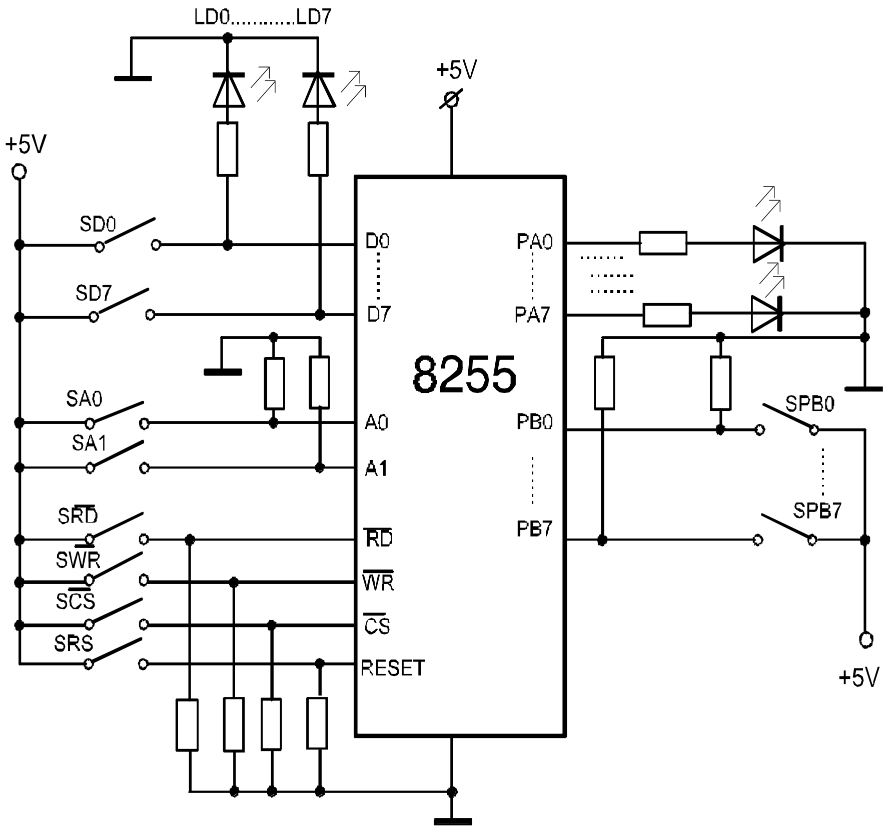
Untuk dapat mengoperasikan PPI 8255 dibutuhkan konfigurasi masukan pada jalur kontrol, dan dengan jalur kontrol inilah penunjukan register kendali, penunjukan port A, port B dan port C dapat dlakukan. Dengan demikian maka operasi baca data dari port atau tulis data ke port dapat dilakukan, dengan A0 dan A1 sebagai penunjuk port dan register kontrol.

Secara rinci konfigurasi fungsi kontrol pada PPI 8255 ditunjukan pada tabel berikut:

Konfigurasi kontrol fungsi PPI 8255.

Reset	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A1	A0	Operasi reset
1	X	X	X	X	X	Port A,B dan C sebagai Masukan
Operasi Membaca (Read)						
0	0	0	1	0	0	Port A → Bus Data
0	0	0	1	0	1	Port B → Bus Data
0	0	0	1	1	0	Port C → Bus Data
Operasi Menulis (Write)						
0	0	1	0	0	0	Bus Data → Port A
0	0	1	0	0	1	Bus Data → Port B
0	0	1	0	1	0	Bus Data → Port C
0	0	1	0	1	1	Bus Data → Register Kontrol
Fungsi yang tidak diperbolehkan						
X	1	X	X	X	X	Bus Data → Berimpedansi tinggi
0	0	0	1	1	1	Kondisi tidak diijinkan
0	0	1	1	X	X	Bus Data → Berimpedansi tinggi

Gambar 3.37 menunjukkan kepada kita bagaimana untuk hubungan masing-masing Pin dan penggunaannya dalam menstransfer data.



Gambar 3.37. Rangkaian uji baca tulis PPI 8255

3. Jalan Operasi Dasar PPI 8255 :

a. <u>Proses Inisialisasi PPI 8255</u>

- Tentukan data kata kendali pada Bus Data (S D7 - S D0)
- Contoh : 82H (Port A=Keluaran, Port B=Masukan).
- SRS → dibuka
- SA0 dan SA1 → ditutup (Alamat Register Kontrol)
- S$\overline{RD}$ → ditutup
- S $\overline{CS}$ → dibuka
- S $\overline{WR}$ → ditutup - dibuka (Operasi Menulis).

Hasil LED PA7 - PA0 = Padam

b. Proses Menulis Data dari Bus Data ke Port A

- Tentukan data (yang akan dikeluarkan ke Port A) pada Bus Data SD7 - SD0
- SR → dibuka
- S A0 dan S A1 → dibuka (alamat Port A)
- S $\overline{RD}$ → ditutup
- A $\overline{CS}$ → dibuka
- S $\overline{WR}$ → ditutup- dibuka (operasi Menulis)

Hasil LED PA7- PA0 menyala sesuai data saklar SD7-SD0 berarti telah terjadi pemindahan data dari bus data ke port A (penulisan data dari Bus Data ke Port A)

c. Proses Membaca Data dari Port B ke Bus Data

- S R → dibuka
- S A0 → ditutup dan S A1 → dibuka (alamat Port B)
- S $\overline{WR}$ → ditutup
- S $\overline{CS}$ → dibuka
- S $\overline{RD}$ → ditutup- dibuka (operasi Membaca)

4. Pengalamatan PPI 8255 pada sistem minimal Z 80

Beberapa pin masukan dan keluaran dari CPU Z 80 dipergunakan dalam peng-alamatan PPI ini. Selain Pin-Pin Kontrol : $\overline{WR}$, $\overline{RD}$, RESET dan Bus Data dari CPU, juga dipergunakan Pin alamat A7-A0 dan Pin $\overline{IOREQ}$.

A7-A0 dipergunakan untuk memberikan data alamat port.

$\overline{IOREQ}$. dipergunakan bersama sinyal-sinyal alamat A7-A2 untuk mengaktifkan PPI 8255.

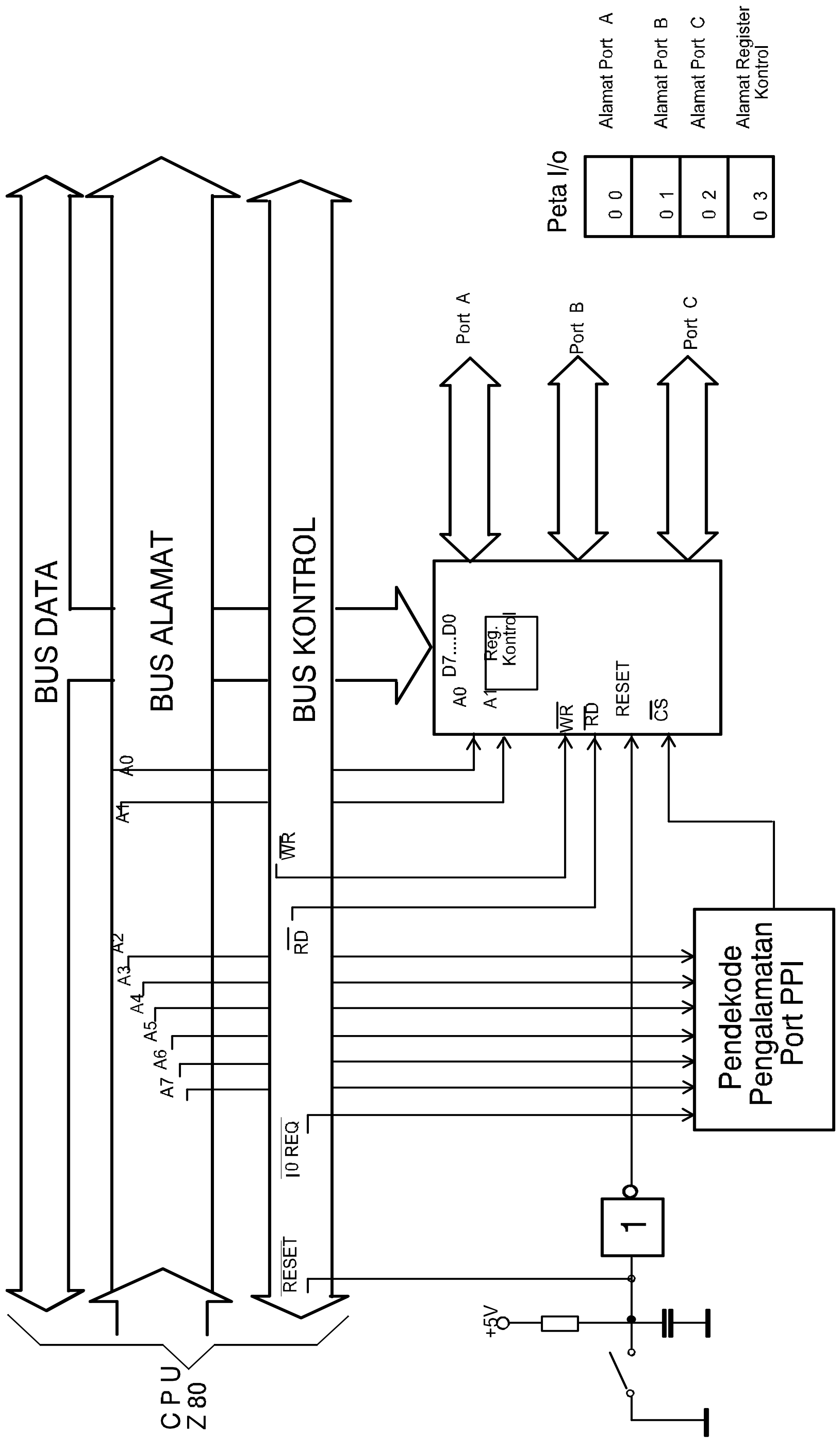
A7-A2 dan $\overline{IOREQ}$. merupakan masukan dari Blok Pendekode Peng-alamatan Port PPI, yang mana dalam Blok Pengalamatan ini dibangun Rangkaian Pendekode. Rangkaian Pendekode ini berfungsi untuk mengaktifkan PPI 8255 pada daerah peng-alamatannya.

Kondisi data A7-A2 (yang bersama-sama $\overline{IOREQ}$., dapat mengaktifkan PPI 8255 melalui $\overline{CS}$) dan kondisi data A1-A0 dapat menentukan/menunjukkan alamat-alamat Port dan Register Kontrol PPI.

Sebelum rangkaian Pendekode/Pengalamatan Port PPI 8255 dibuat, maka kita harus menentukan peta alamat Port masukan keluaran.

Dasar penentuan alamat port input/output menggunakan PPI 8255 adalah dengan menetukan terlebih dahulu alamat yang akan digunakan, sebagai contoh ditentukan seperti pada peta lokasi port berikut:

	Port A ------------ 00H
	Port B ------------ 01H
	Port C ------------ 02H
	Register Kontrol ----- 03H


BUS DATA
BUS ALAMAT
BUS KONTROL
A0
A1
A2
A3
A4
A5
A6
A7
WR
RD
IO REQ
RESET
D7....D0
A0
A1
Reg. Kontrol
WR
RD
RESET
CS
Port A
Port B
Port C
Peta I/o
0 0
0 1
0 2
0 3
Alamat Port A
Alamat Port B
Alamat Port C
Alamat Register Kontrol
Pendekode Pengalamatan Port PPI
1
+5V
C P U
Z 80

Gambar 3.38. Blok rangkaian aplikasi PPI 8255

Peta port berdasarkan data yang telah ditentukan tersebut di atas perlu ditindak lanjuti dengan rangkaian pengalamatan secara hardware, seperti diketahui bahwa pada *chip* IC PPI 8255 terdapat A0 dan A1 yang digunakan sebagai saluran penunjuk port sehingga jika diinginkan dengan alamat tersebut maka hanya diperlukan kombinasi dengan sinyal baca/tulis dan permintaan jalur *input/output*.

Secara diagram blok sistem pengalamatan PPI 8255 ditunjukan pada gambar 3.38, yang di dalamnya terdapat PPI 8255, dekoder sebagai penunjuk alamat PPI yaitu untuk port A adalah 00, untuk port B adalah 01, dan port C adalah 02.

Untuk mengendalikan fungsi dari port A, port B dan port C yaitu untuk fungsi input atau fungsi output dilakukan dengan men-set register kontrol, sedangkan register kontrol adalah alamat tertinggi yaitu menempati alamat 03. Berikut gambar sistem pengalamatan port PPI 8255:

Jika alamat yang digunakan adalah bus alamat A0, A1, A2, A3, A4, A5, A6 dan A7 maka dengan menyertakan permintaan port input/utput $\overline{IOREQ}$ dapat ditabelkan seperti tabel berikut:

Penentuan alamat port menggunakan abus alamat A0 …A7

$\overline{IOREQ}$	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	1

Penunjuk alamat Chip IC

Penunjuk alamat port

A0 dan A1 merupakan jalur penunjuk alamat port yang terdiri dari port A, port B, port C dan register kendali dalam IC PPI 8255, sedangkan A2, A3, A4, A5,

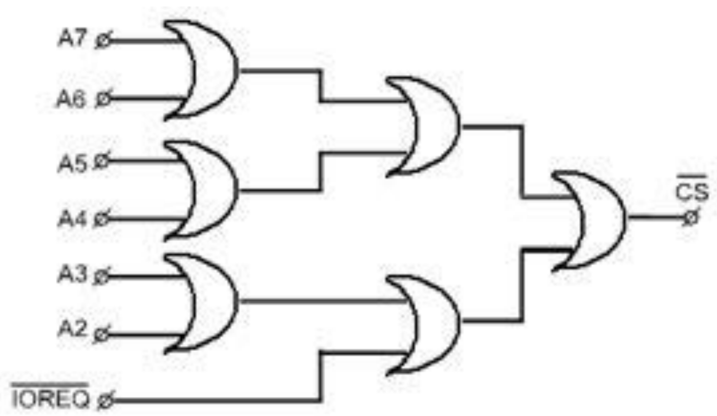
A6 dan A7 digunakan untuk menunjuk alamat chip IC dalam hal ini ditentukan basis alamat 00. Untuk mengaktipkan chip IC PPI 8255 digunakan kontrol permintaan I/O yaitu melalui jalur kontrol $\overline{IOREQ}$

Dari bantuan tabel diatas, kita dapat menganalisa, bahwa untuk mengaktifkan PPI 8255, kondisi A7-A2 dan $\overline{IOREQ}$.dari CPU harus berkondisi "LOW" ("0"). dan untuk menentukan alamat Port A,B,C dan Register Kontrol ditentukan oleh A1 dan A0.

Untuk membangun rangkaian Pendekode Pengalamatan Port sesuai data hasil analisa diatas,dapat dibangun dengan mempergunakan gerbang TTL dasar atau dengan Dekoder TTL 74138/74139.

Dengan memanfaatkan 6 (enam) buah gerbang dasar OR dan dengan masukan berupa bus alamat A0 sampai A7 serta permintaan jalur input/output, maka rangkaian Pendekode Pengalamatan untuk menunjuk dan mengaktifkan Port PPI seperti ditunjukan oleh gambar 3.39.

Hasil dari pengkodean pengamalatan PPI 8255 digunakan untuk menunjuk atau memilih chip IC yang ingin diaktifkan, dalam hal ini untuk memberikan sinyal pada chip select (pemilih chip) $\overline{CS}$.



Gambar 3.39. Pendekodean Pengalamatan Port PPI

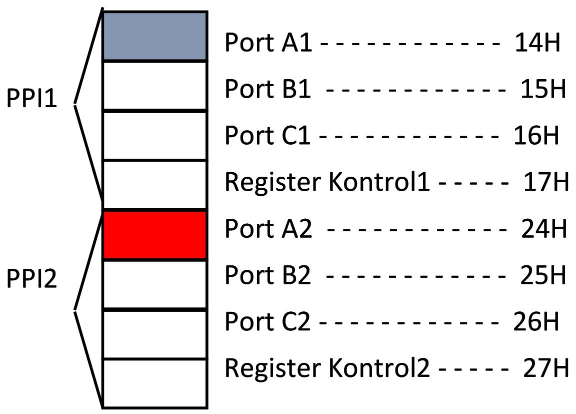
Hubungan - hubungan PIN dari PPI dengan CPU pada sistem Mikroprosessor dengan PPI lebih dari 1 adalah seperti pada sistem Mikroprosessor dengan 1

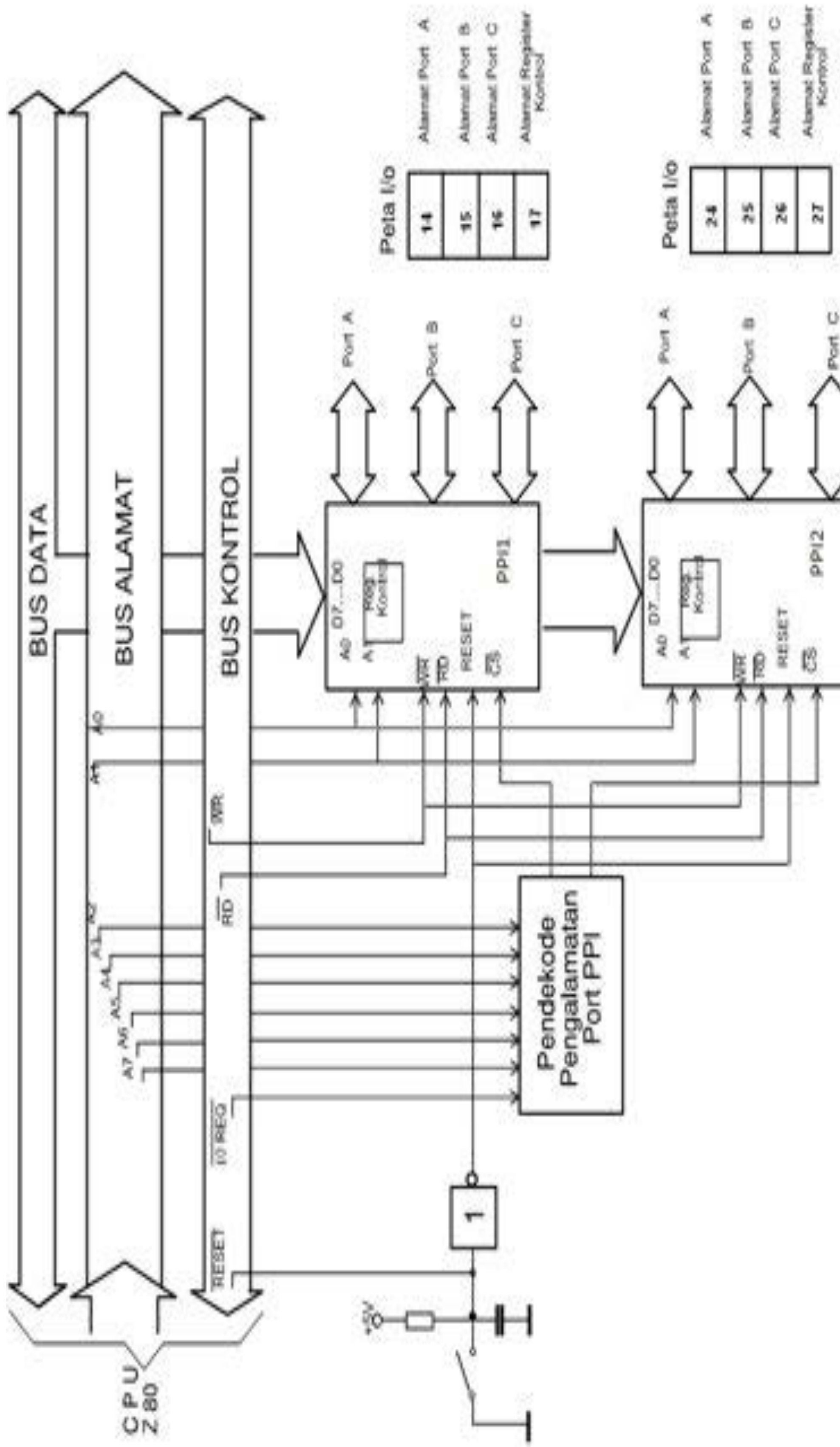
PPI, walaupun demikian terdapat perbedaan yaitu pada rangkaian Pendekode Pengalamatan Port PPI.

Sebagai contoh penggunaan A0 sampai A7 sebagai berikut:

IOREQ	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	1	0	1	0	0
0	0	0	0	1	0	1	0	1
0	0	0	0	1	0	1	1	0
0	0	0	0	1	0	1	1	1
0	0	0	1	0	0	1	0	0
0	0	0	1	0	0	1	0	1
0	0	0	1	0	0	1	1	0
0	0	0	1	0	0	1	1	1

Berdasarkan susunan alamat tersebut membentuk alamat seperti berikut:



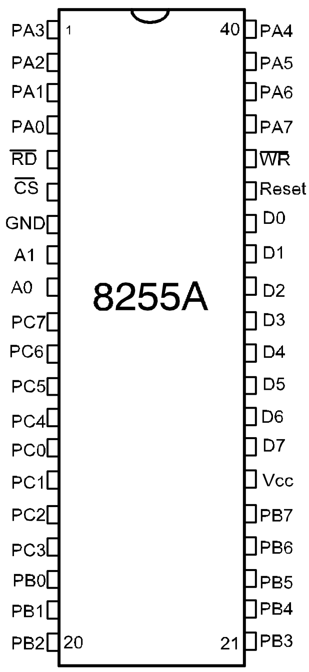

BUS DATA
BUS ALAMAT
BUS KONTROL
CPU
Z 80
PPI1
PPI2
Port A
Port B
Port C
RESET
Pendekode
Pengalamatan
Port PPI
Peta I/o
14
15
16
17
24
25
26
27
Alamat Port A
Alamat Port B
Alamat Port C
Alamat Register
Kontrol

Gambar 3.40. Rangkaian pengalamatan 2 buah port PPI 8255

Latihan

1. Jelaskan fungsi Pin - Pin dari Konfigurasi PPI 8255 !

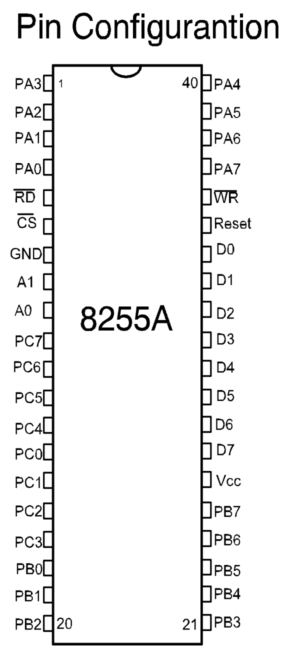
Pin Configurantion



2. Apa fungsi $\overline{CS}$ (Chip Select), $\overline{IOREQ}$., $\overline{WR}$, $\overline{RD}$

Jawaban

1. Jelaskan Pin - Pin dari Konfigurasi PPI 8255 dengan fungsinya masing-masing.



Nama Pin

D7 - D0	Data Bus (Bi-Directional)
RESET	Reset Input
CS	Chip Select
RD	Read Input
WR	Write Input
A0,A1	Port Adress
PA7-PA0	Port A (BIT)
PB7-PB0	Port B (BIT)
PC7-PC0	Port C (BIT)
Vcc	+5 Volts
GND	0 Volt

Deskripsi pin PPI 8255

a. Pin - Pin Saluran Data :

- Bus Data : D7 - D0
- Bus Port A : PA7 - PA0
- Bus Port B : PB7 - PB0
- Bus Port C : PC7 - PC0

b. Pin - Pin Saluran Pengontrol :

- Baca (Read) : $\overline{RD}$
- Tulis (Write) : $\overline{WR}$
- Reset

2. Apa fungsi $\overline{CS}$ (Chip Select)

$\overline{CS}$ (Chip Select) sebagai pemilih Chip :

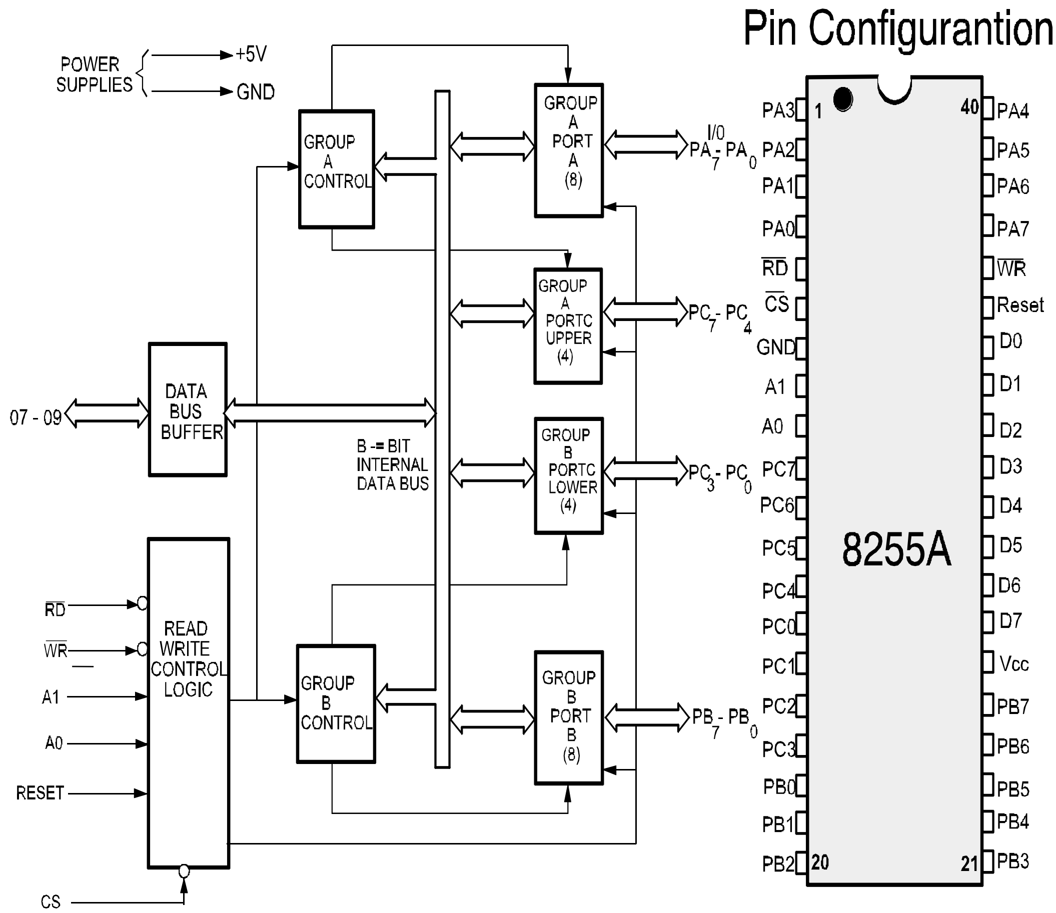
Untuk mengaktifkan Chip, bekerja dengan Logika Low (“0”).

5. Aplikasi Fungsi PPI 8255

a. Konfigurasi Internal PPI 8255

IC PPI 8255 adalah peranti perantara pheriperal terprogram yang di desain untuk kegunaan dalam sistem Mikrokomputer. Fungsinya adalah sebagai komponen Multiguna masukan ataupun keluaran. Untuk perantara antara peralatan pheriperal luar dengan sistem mikrokomputer.

Konfigurasi Fungsi 8255 diprogram oleh sistem software tertentu tidak lepas dari pemahaman tentang rangkaian internalnya, berikut merupakan penjelasan fungsi setiap bagian sistem PPI 8255. Secara blok diagram ditunjukan pada gambar x berikut:



Gambar 3.41. Diagram blok internal PPI 8255

Buffer Bus Data :

Buffer 8 bit dua (2) arah tiga (3) state ini, dipergunakan sebagai perantara 8255 dengan bus data sistem.

Data diterima atau dikirim oleh buffer tergantung perintah masukan atau keluaran oleh CPU.

Informasi kata kendali dan status dikirim melalui buffer.

Baca Tulis dan Logik Kontrol

Fungsi dari blok ini adalah untuk mengatur semua pengiriman internal dan external dari data dan kata kendali atau kata status.

8255 menerima masukan dari bus alamat dan bus kontrol CPU dan memfungsikannya untuk pelaksanaan tugas masing-masing kelompok kontrol 8255.

$\overline{CS}$

Chip Select (pemilih chip) kondisi “LOW” pada pin input ini, mengijinkan terjadinya komunikasi antara 8255 dengan CPU.

$\overline{RD}$

Read (pembacaan) kondisi “LOW” pada pin input ini, mengijinkan 8255 untuk mengirimkan informasi data ke CPU melalui Bus data.

Pada prinsipnya mengijinkan CPU membaca informasi data dari 8255.

$\overline{WR}$

Write (penulisan) kondisi “LOW” pada pin input ini, memungkinkan CPU untuk menulis informasi data ke 8255.

A_0 - A_1

Pemilih Port. Signal input-input ini, mengontrol pemilihan satu (1) dari empat (4) Port : Port A,B,C dan Register Kontrol. Ini biasanya dihubungkan dengan Bit - bit LSB dari bus alamat (A0 dan A1).

Fungsi kendali PPI 8255

A1	A0	RD	WR	CS	Input Operation (READ)
0	0	0	1	0	
0	1	0	1	0	
1	0	0	1	0	
					Output Operation (WRITE)
0	0	1	0	0	
0	1	1	0	0	
1	0	1	0	0	
1	1	1	0	0	
					Disable Function
X	X	X	X	1	
1	1	0	1	0	
X	X	1	1	0	

Berdasarkan tabel di atas terdapat 2(dua) fungsi utama yang diijinkan untuk operasi PPI 8255, yaitu baca data dari devais luar ke mikroprosesor yang berarti operasi input dan tulis data dari mikroprosesor ke devais luar berarti operasi output. Sedangkan fungsi ke tiga tidak diijinkan, oleh karena itu fungsi ini disable.

Reset

Reset kondisi “HIGH” pada input ini, akan menghapus isi Register Kontrol dan semua Port (A,B, dan C) dan semua Port di “SET” sebagai masukan.

b. Sistem Pengontrolan Port

Konfigurasi fungsi dari tiap-tiap “PORT” diprogram oleh software sistem, yang pada prinsipnya CPU mengirimkan data kata kendali ke register kontrol 8255. Kata kendali berisikan informasi seperti “MODE”, “SET BIT”,”RESET BIT” dan seterusnya, yang akan meng-inisialisasi konfigurasi fungsi dari port 8255.

Setiap kelompok kontrol (untuk kelompok A dan B) menerima perintah dari “Logik Kontrol Read/Write”, menerima kata kendali dari bus data internal dan mengijinkan untuk pembentukan hubungan/pengelompokan port dan fungsinya.

- Kelompok kontrol A - Port A dan Port C atas (C7 - C4).
- Kelompok kontrol B - Port B dan Port C bawah (C3 - C0).
- Register kata kendali hanya dapat ditulis.

Operasi pembacaan pada register kata kendali tidak diijinkan.

Port A,B dan C

8255 terdiri dari 3 port 8 bit (A,B dan C), semua dapat dikonfigurasikan (dalam bermacam-macam fungsi) oleh Soft Ware Sistem.

Port A

Sebuah Buffer/Penyimpan keluaran data 8 bit dan sebuah penyimpan masukan data 8 bit.

Port B

Sebuah Buffer/Penyimpan keluaran data 8 bit dan sebuah penyimpan masukan data 8 bit.

Port C

Sebuah Buffer/Penyimpan keluaran data 8 bit dan sebuah Buffer masukan data 8 bit. Port ini dapat dibagi menjadi 2 port 4 bit melalui pengaturan Mode Kontrol. Setiap Port 4 bit terdiri dari sebuah penyimpan 4 bit dan itu digunakan untuk keluaran sinyal kontrol dan masukan sinyal status.

c. Proses Operasi 8255

- **Pemilihan Mode**

Ada 3 mode, dasar operasi yang dapat dipilih oleh Software sistem :

- Mode 0 - Dasar masukan dan keluaran
- Mode 1 - Masukan/Keluaran sesuai sinyal isyarat
- Mode 2 - Bus dua arah

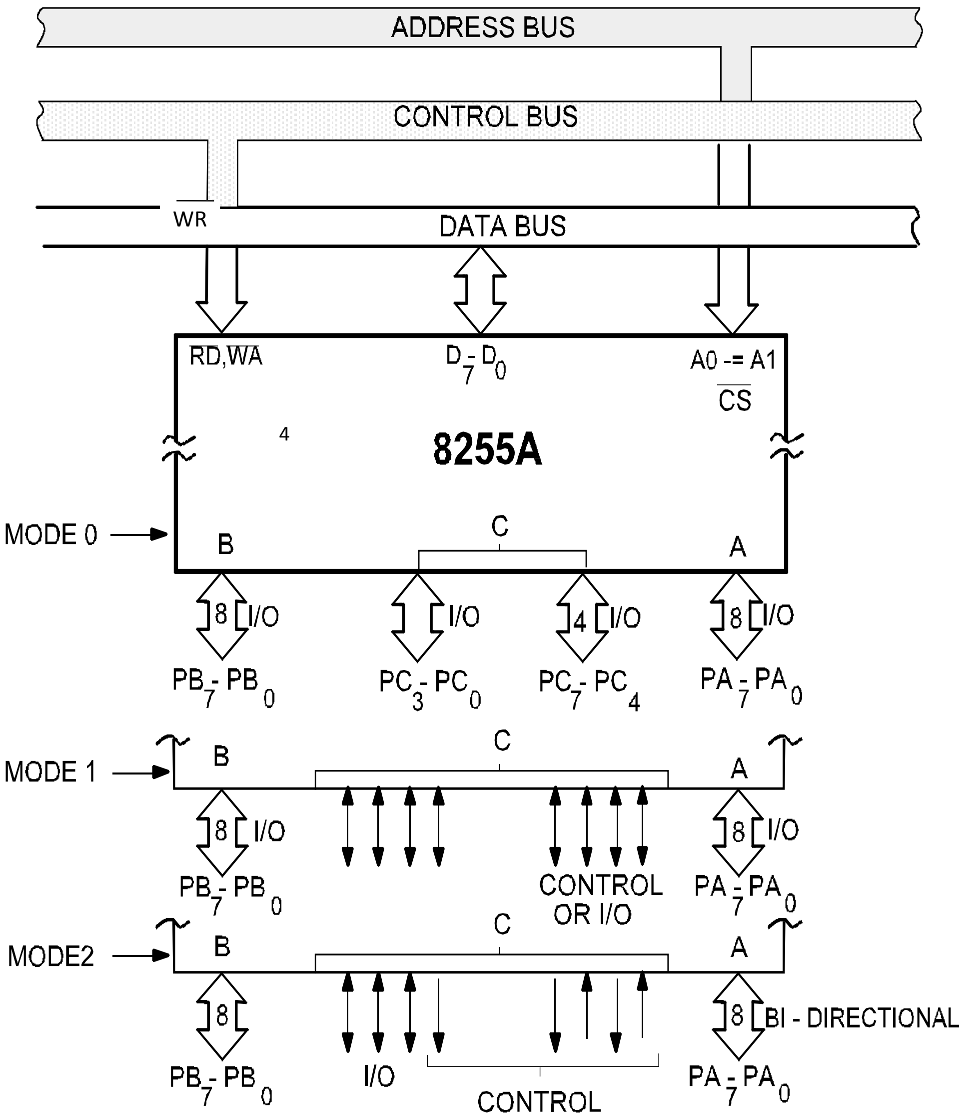
Bila masukan RESET menuju “High”, semua port akan di set menjadi Mode masukan (keadaan berimpedansi tinggi). Mode untuk port A dan port B dapat ditentukan secara terpisah saat port C dibagi dalam 2 bagian sesuai yang ditentukan oleh pendefinisian port A dan port B.

Semua register keluaran termasuk flip-flop status akan direset bila mode diganti.

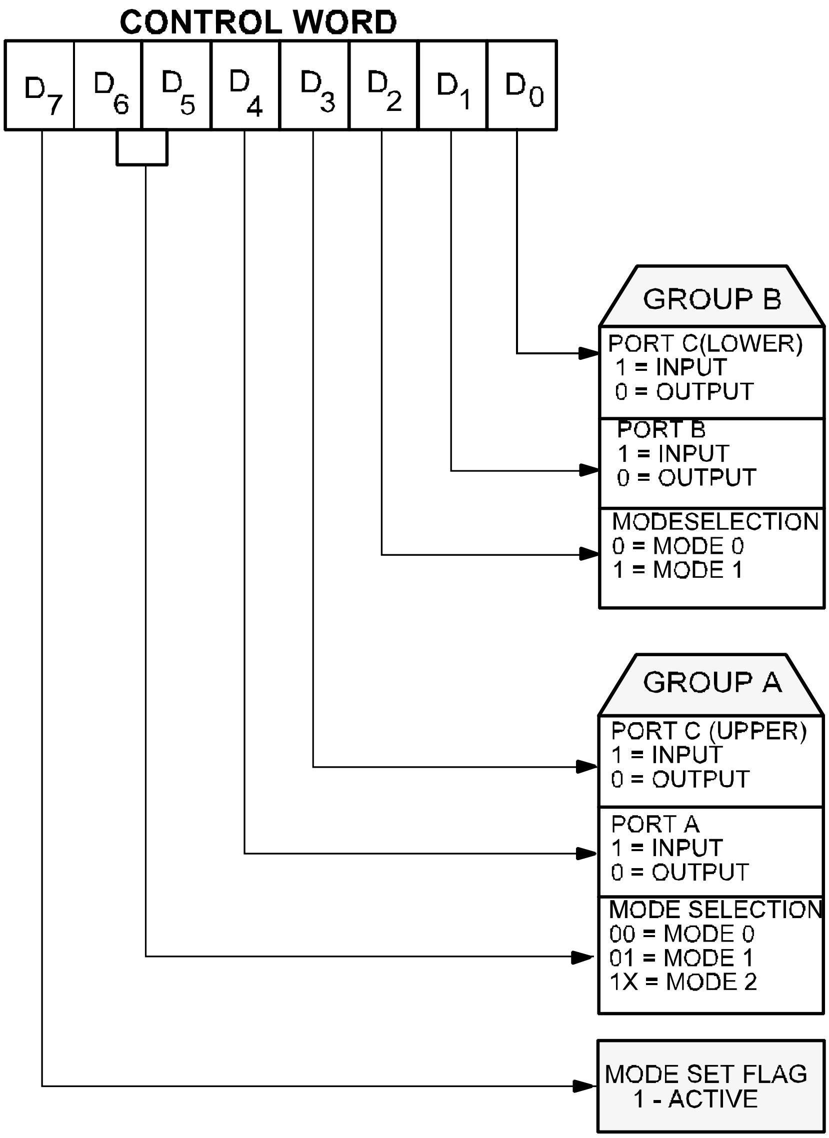
- **Bit Tunggal untuk Set - Reset**

Beberapa bit dari 8 bit pada port C dapat di-Set atau di-Reset dengan menggunakan perintah Out. keistimewaan ini mengurangi Soft Ware, yang digunakan dalam aplikasi yang berdasarkan kontrol.

Bila port C digunakan sebagai status/control untuk Port A atau Port B, Bit ini dapat di-Set/Reset oleh penggunaan operasi Set/Reset bit. Sebagaimana jika digunakan sebagai port data keluaran.



Gambar 3.42. Definisi Mode Dasar dan Bus Perantara



Gambar 3.43. Definisi fungsi PPI 8255 melalui format Mode

- **Mode Operasi**

Mode 0 :

Konfigurasi fungsi menyediakan operasi masukan/keluaran yang sederhana untuk masing-masing port. Tidak mengharuskan ada “HandShaking” (pertukaran isyarat dari dua peranti yang saling berhubungan), data secara sederhana ditulis ke atau dibaca dari port tertentu.

Definisi dari fungsi dasar Mode 0 :

⇒ Dua (2) port 8 bit dan dua (2) port 4 bit

⇒ Setiap port dapat sebagai masukan atau keluaran

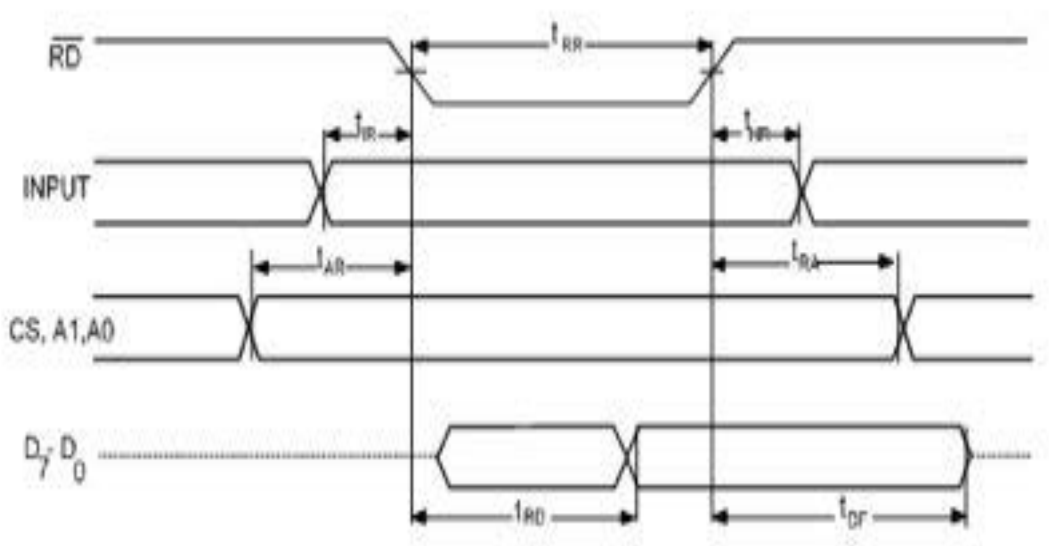
⇒ Keluaran di simpan

⇒ Masukan tidak disimpan

⇒ Memungkinkan 16 jenis konfigurasi masukan/keluaran

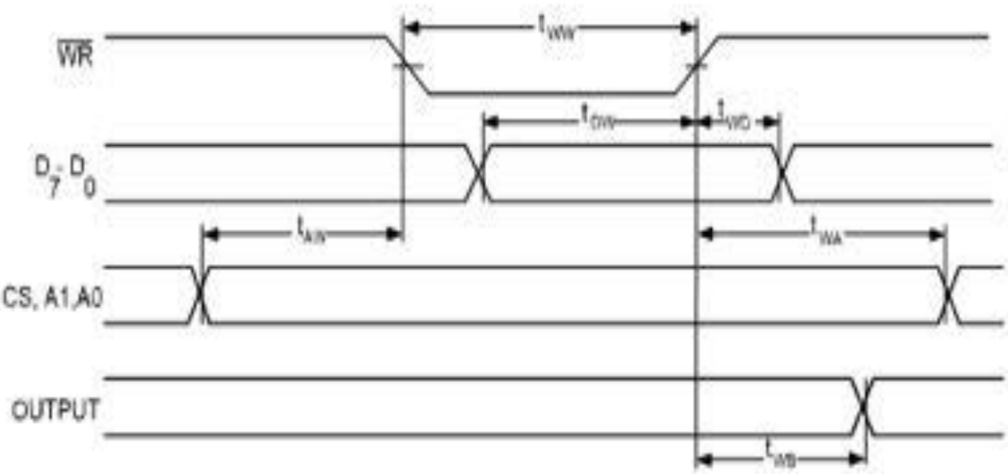
Perhatikan gambar 6 : Definisi port untuk Mode 0

MODE 0 (BASIC INPUT) → Proses Sinyal Baca pada Mode 0



Gamabar 3.44. Diagram pulsa proses baca port fungsi input

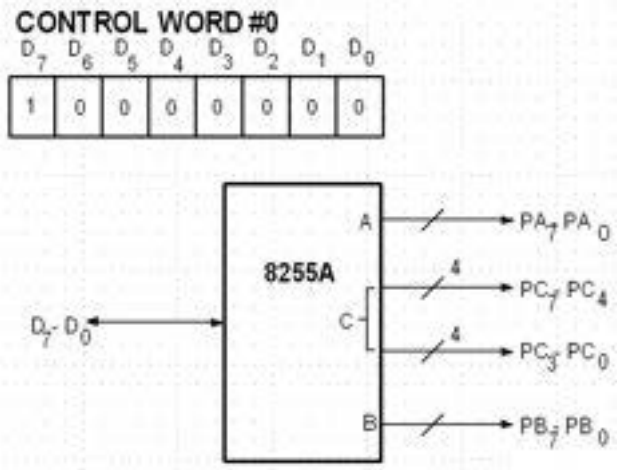
MODE 0 (BASIC OUTPUT) → Proses Sinyal Tulis pada Mode 0



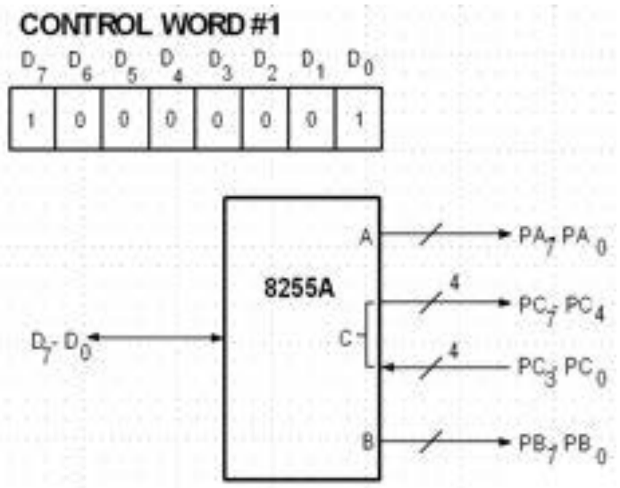
Gamabar 3.45. Diagram pulsa proses tulis port fungsi input

Definisi Port MODE 0

A		B		GROUP A		#	GROUP B	
D4	D3	D1	D0	Port A	Port C (Upper)		Port B	Port C (Lower)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

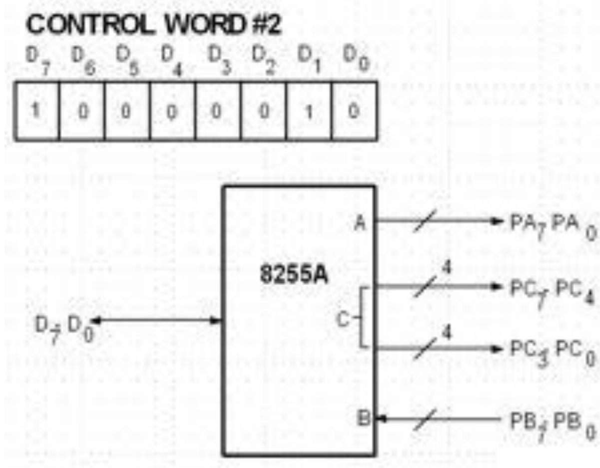


Dengan pengisian kata kendali 80H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai output dan port C juga berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan port C.

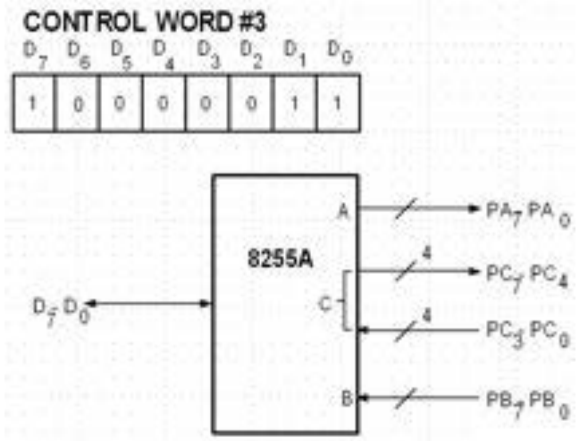


Dengan pengisian kata kendali 81H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai output dan port C0-3 berfungsi sebagai input dan port C4-7 berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan port C4-7 dan baca melalui port C0-3.

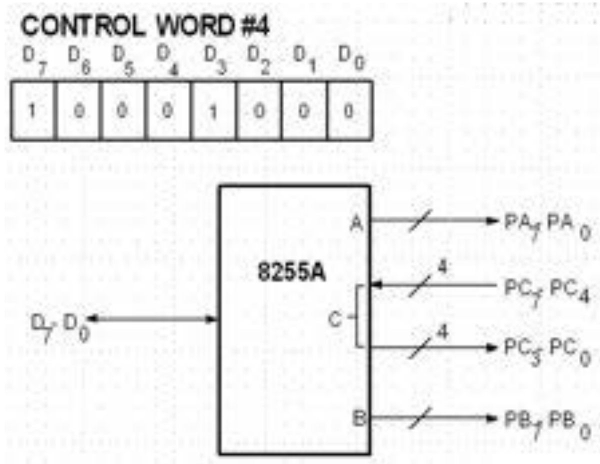




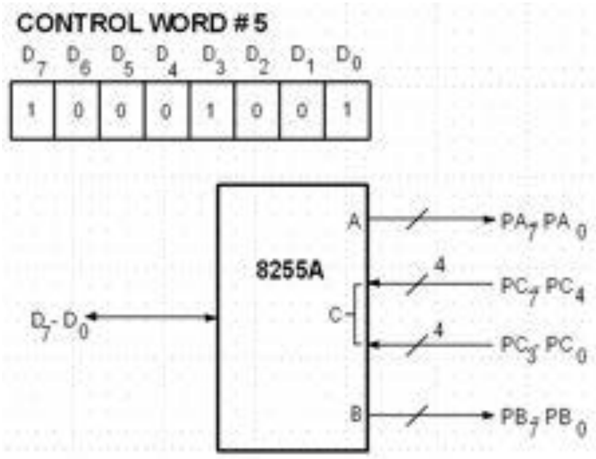
Dengan pengisian kata kendali 82H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai input dan port C juga berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port C dan baca melalui port B.



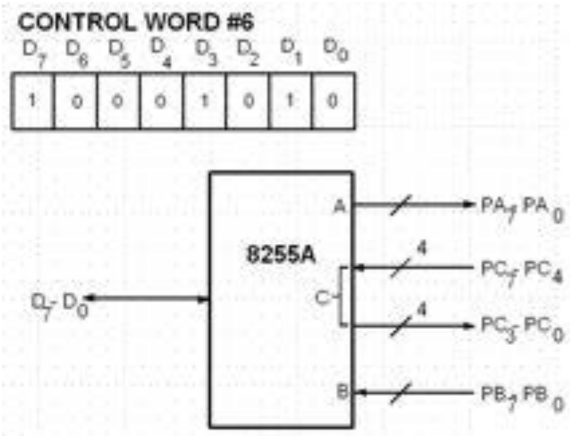
Dengan pengisian kata kendali 83H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai input dan port C0-3 berfungsi sebagai input dan port C4-7 berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A dan port C4-7, untuk baca melalui port B dan port C0-3.



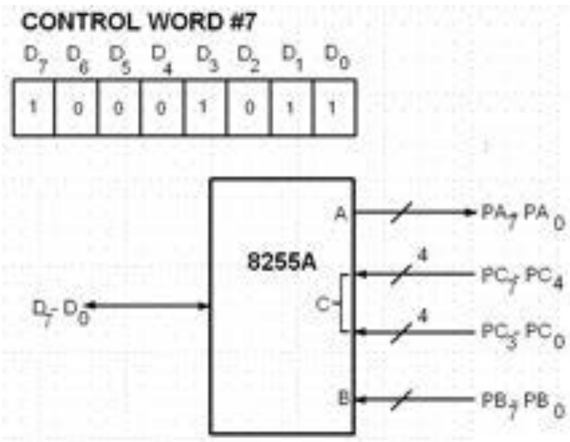
Dengan pengisian kata kendali 88H ke dalam register kontrol, maka port A berfungsi output, port B berfungsi output dan port C0-3 berfungsi output. dan port C4-7 berfungsi input. Proses tulis dilakukan melalui port A, B dan port C0-3, untuk baca melalui port C4-7.



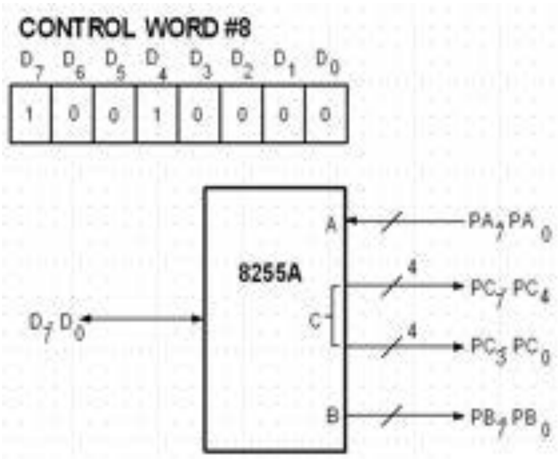
Dengan pengisian kata kendali 89H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai output dan port C berfungsi input. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan baca melalui port C.



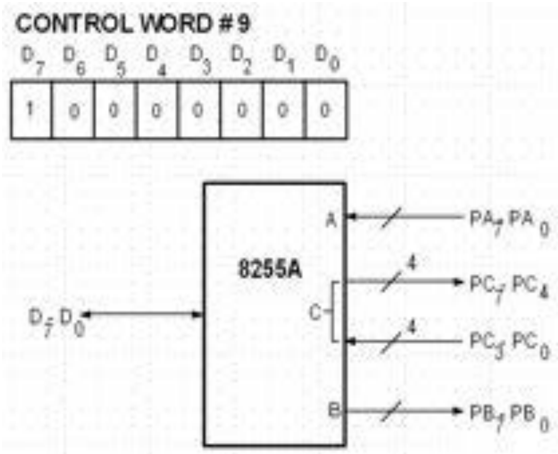
Dengan pengisian kata kendali 8AH ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai output dan port C juga berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan port C.



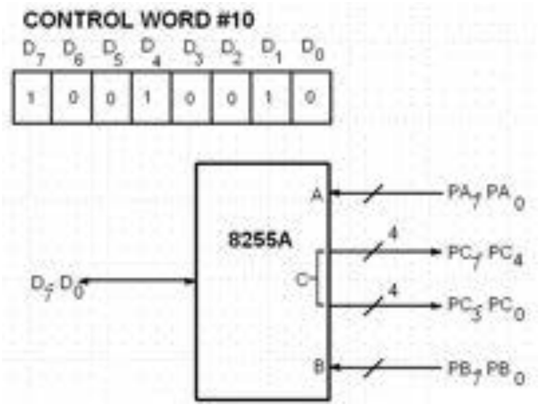
Dengan pengisian kata kendali 8BH ke dalam register kontrol, maka port A berfungsi sebagai output, port B dan port C berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, baca melalui port B dan port C.



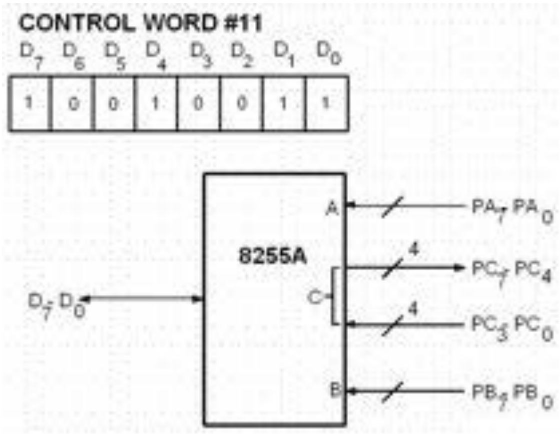
Dengan pengisian kata kendali 80H ke dalam register kontrol, maka port A berfungsi sebagai output, port B berfungsi sebagai output dan port C juga berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan port C.



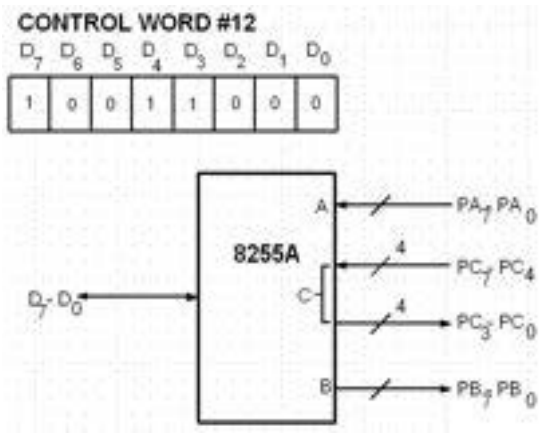
Dengan pengisian kata kendali 90H ke dalam register kontrol, maka port A dan port C0-3 berfungsi sebagai input, port B dan port C4-7 sebagai output. Melalui konfigurasi ini proses baca melalui port A dan C0-3, tulis melalui port B dan port C4-7.



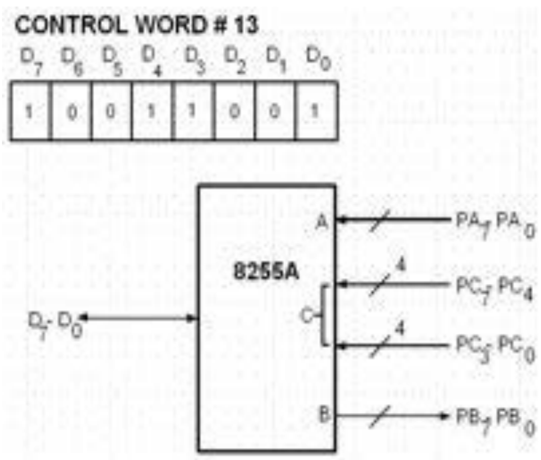
Dengan pengisian kata kendali 92H ke dalam register kontrol, maka port A dan port B berfungsi sebagai input, dan port C berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A dan port B dan tulis melalui port C.



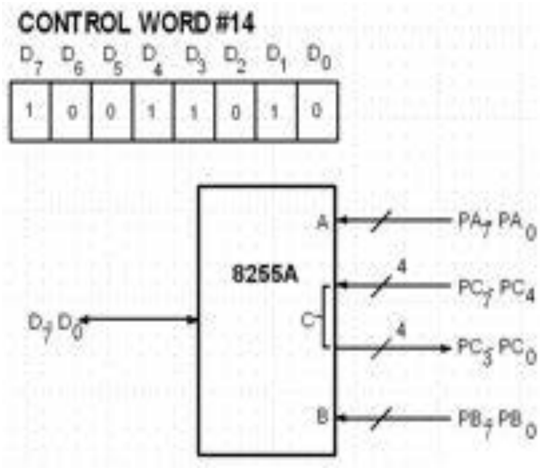
Dengan pengisian kata kendali 93H ke dalam register kontrol, maka port A port B dan port C0-3 berfungsi sebagai input, port C4-7 berfungsi sebagai output. Melalui konfigurasi ini proses baca dapat dilakukan melalui port A, port B dan port C0-3 dan tulis melalui C4-7..



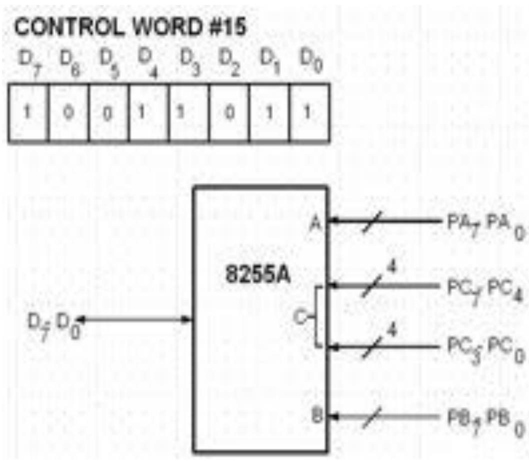
Dengan pengisian kata kendali 98H ke dalam register kontrol, maka port A dan port C4-7 berfungsi sebagai input, port B dan port C0-3 berfungsi sebagai input. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port B dan port C0-3, baca melaui port A dan port C4-7..



Dengan pengisian kata kendali 99H ke dalam register kontrol, maka port A dan port C berfungsi sebagai input, port B berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port B, dan baca melalui port A dan port C.



Dengan pengisian kata kendali 9AH ke dalam register kontrol, maka port A, port B dan port C4-7 berfungsi sebagai input, port C0-3 berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port C0-3, baca melalui port A, port B dan port C4-7.



Dengan pengisian kata kendali 80H ke dalam register kontrol, maka port A, port B dan port C berfungsi sebagai output. Melalui konfigurasi ini proses tulis dapat dilakukan melalui port A, port B dan port C.

Gambar 3.46. Fungsi Kata kendali penentu Input/Output

Latihan

1. Ada 3 Mode dasar operasi yang dapat dipilih oleh Soft Ware Sistem. Sebutkan dan Jelaskan masing - masing !

2. Pada Mode 0 : jika D0 = 1 D3 = 1

   D1 = 0 D4 = 0

Bagaimana Fungsi daripada Port A

Port B

Port C tinggi

Port C rendah

3. Jika awal program, pada monitor tertulis :

```
LD    A, 92 H
Out   (FBH), A
```

Apa arti Op - code diatas.



Jawaban

1. Ada 3 mode dasar operasi yang dapat dipilih oleh Soft Ware sistem :
    * Mode 0 → Dasar masukan dan keluaran
    * Mode 1 →Masukan keluaran sesuai sinyal isyarat
    * Mode 2 → Bus dua arah

2. Pada mode 0, jika D4 = 0 D3 = 0 D1 = 0 D0 = 1

maka Port A sebagai = Output (keluaran)

Port C tinggi sebagai = Input (masukan)

Port B sebagai = Output (keluaran)

Port C rendah sebagai = Input (masukan)

3. Jika awal program adalah

```
LD   A, 92 H
Out
```

Artinya : Merupakan kedudukan awal dari PPI 8255. Dengan demikian kontrol word 92 H dikirim ke kontrol word register, sehingga fungsi dari masing-masing port ;

1	0	0	1	0	0	1	0

Mode 0

Port A → Input

Port B → Input

Port C → Output

→ Output

3.20. Perencanaan Minimal Sistem Z-80

Minimal sistem sebuah komputer atau bisa disebut juga komputer sederhana menggunakan mikroprosesor Z80 dapat dibangun dengan menggunakan komponen dasar mikroprosesor Z80, memori RAM 6116, EPROM 2716, parallel port interface (PPI) 8255 dan beberapa komponen pendukung clock berikut sistem pengalamatam.

Sistem perencanaan dan pembuatan serta sistem kerja dari pada minimal sistem Z80 dapat dilakukan melalui beberapa langkah yaitu meliputi:

- Merencanakan dan menempatkan blok CPU Z - 80
- Merencanakan dan menempatkan sistem kontrol CPU Z - 80
- Merencanakan dan menempatkan saluran data
- Merencanakan dan menempatkan saluran alamat
- Merencanakan dan menempatkan blok memori
- Merencanakan dan menempatkan blok input/output
- Merencanakan dan menempatkan sistem kontrol untuk memori (RAM dan EPROM)
- Merencanakan dan menempatkan sistem kontrol untuk input/output (8255)

Rangkaian minimal sistem yang kita rencanakan dibawah ini meliputi :

1. Blok CPU Z - 80
2. Blok penyimpan
3. Blok masukan keluaran (input - Output)

Blok penyimpan data program dan blok masukan keluaran telah kita bahas pada pengalamatan penyimpan RAM - EPROM pengalamatan PPI 8255

1. Blok CPU Z - 80

Dalam membuat rangkaian minimal sistem Z-80 perlu diperhitungkan rangkaian internalnya, pin koneksi yang berfungsi sebagai penghubung internal mikro-prosesor dengan eksternal mikroproseso. Seperti dijelaskan



pada paragraf sebelumnya bahwa sebuah mikroprosesor Z80 dapat dihubungkan dengan peralatan luar mikroprosesor melalui ketiga bus, yang terdiri dari :

⇒ Bus data, yang berfungsi sebagai penghubung data pada setiap bagian internal mikroprosesor dan eksternal melalui bus data (D0 sampai D7) melalui pin seperti ditunjukan pada gambar konfigurasi pin mikroprosesor Z-80.

⇒ Bus alamat, berfungsi sebagai penghubung penunjukan alamat baik memori maupun port Input/output, terutama untuk RAM/EPROM dan port I/O di luar sistem mikroprosesor.

⇒ Bus kontrol, berfungsi sebagai kendali proses baca/tulis untuk data yang melewati devais eksternal, untuk RAM/EPROM dan port I/O di luar sistem mikroprosesor

Bus kontrol terdiri dari :

- Kontrol sistem
- Kontrol CPU
- Kontrol Bus CPU

Dalam rangkaian minimal sistim Z - 80, tidak semua pin dari CPU Z - 80 dipergu- nakan melainkan hanya beberapa pin/fungsi yang kita pergunakan antara lain.

a. Bus data (D1 - D0) , Input Output

Bus ini dipakai sebagai penghantar data 8 bit baik yang dari CPU ke penyimpan data (memori) atau masukan keluaran (Input - Output), atau sebaliknya. Arah jalannya sinyal pada bus ini, masuk ke CPU atau keluar dari CPU.

b. Bus alamat (A15 - A0), output

Dengan bantuan bus alamat ini, CPU dapat memilih lokasi penyimpan data/memori atau lokasi register dari masukan/keluaran (Input Output) yang berbeda-beda. Arah jalannya sinyal pada bus ini, keluar dari CPU.

c. Bus kontrol

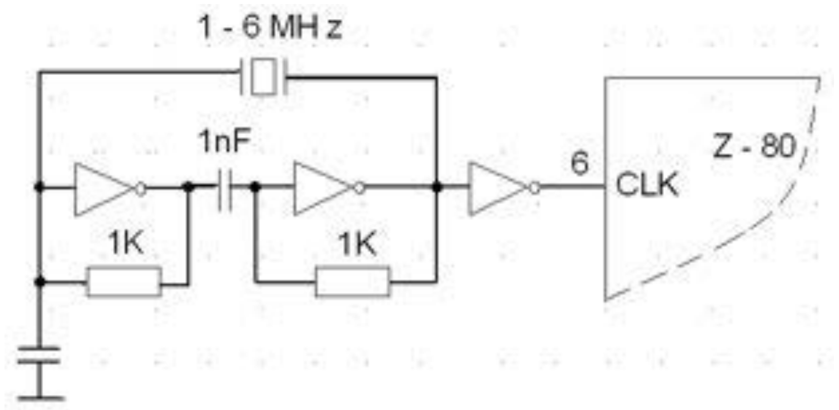
Terdiri dari 13 saluran, yang arahnya masuk atau keluar ke / dari CPU. Masing sinyal kontrol ini dijelaskan sebagai berikut :

2. Saluran detak (CLK), Input :

Sinyal detak diumpankan ke CPU melalui saluran CLK. Dengan demikian CPU dan juga bus sistim akan bekerja mengikuti sinyal detak.

CPU Z - 80 dapat bekerja baik dengan frekuensi detak 1 MHz - 6 MHz.

Rangkaian pembangkit detak :

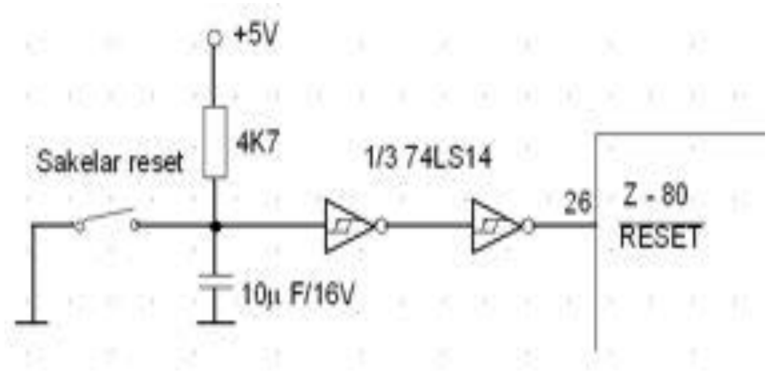


Gambar 3.47. Rangkaian pembangkit pulsa clock

a. Saluran Reset (RESET) , Input :

Sinyal “ LOW “ diberikan pada saluran ini, maka penghitung Program (PC) akan diset dengan nilai 0000H dan Interupt Enable (Pengaktif Interupt) akan direset.

Bila kemudian sinyal “ High “ diberikan pada saluran ini, maka program akan berjalan mulai dari alamat penyimpan program 0000H (Start awal)

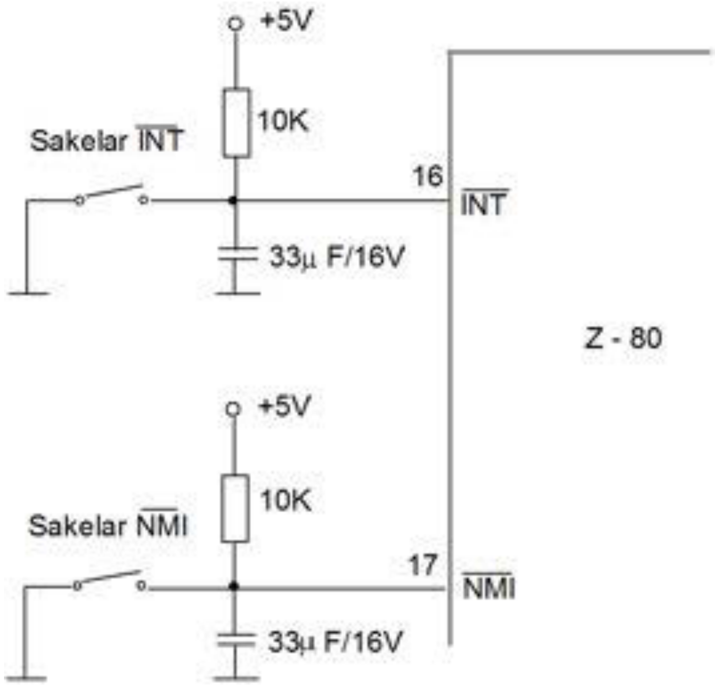


Gambar 3.48. Rangkaian pembangkit Input Reset

b. Saluran Penyela ($\overline{\text{INT}}$ dan $\overline{\text{NMI}}$)

Input Sinyal “ Low “ yang diberikan pada saluran $\overline{\text{INT}}$, memungkinkan untuk melaksanakan salah satu dari 3 penyelaan yang tersebunyi yaitu dalam modus 0, 1 dan 2.

Sedang sinyal “ Low “ pada saluran $\overline{\text{NMI}}$, CPU akan menjalankan program bagian yang berada pada alamat penyimpan program (RAM EPROM) 0066H.



Gambar :3.49. Rangkaian Pembangkit Sinyal $\overline{INT}$ dan $\overline{NMI}$

c. Saluran $\overline{MREQ}$, Output

Apabila CPU membaca op-code (kode operasi) yang berhubungan dengan alamat penyimpanan data/program (memori), maka CPU akan mengaktifkan saluran $\overline{MREQ}$, $\overline{MREQ}$ = “Low“.

Sinyal $\overline{MREQ}$ ini dipakai bersama dengan data alamat (A_{15}-A_0) untuk menunjuk alamat memori yang diinginkan.

d. Saluran $\overline{IOREQ}$, Output

Apabila CPU membaca op code yang berhubungan dengan unit masukan keluaran (PI0), maka CPU akan mengaktifkan sinyal pengontrol masukan keluaran (Input - Output) $\overline{IOREQ}$, $\overline{IOREQ}$ = “ Low “.Sinyal $\overline{IOREQ}$ dipakai bersama dengan data alamat (A_7-A_0) untuk menunjuk alamat Port Input Output yang diinginkan.

e. Saluran $\overline{RD}$, Output

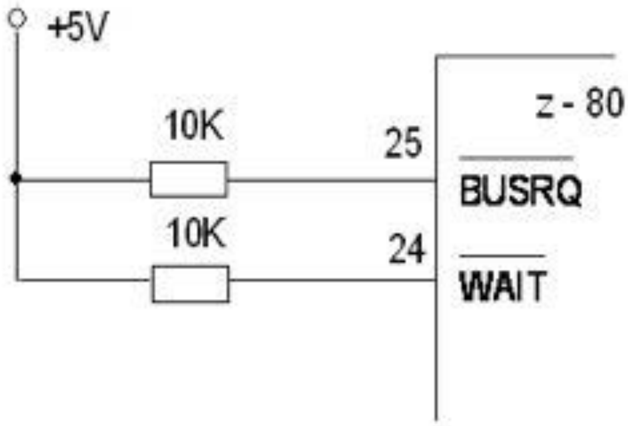


CPU akan mengaktifkan saluran $\overline{RD}$, $\overline{RD}$ = " Low ", selama CPU melaksanakan perintah untuk membaca data dari lokasi penyimpanan data/program (memori) atau register Input Output (port).

f. Saluran $\overline{WR}$, Output

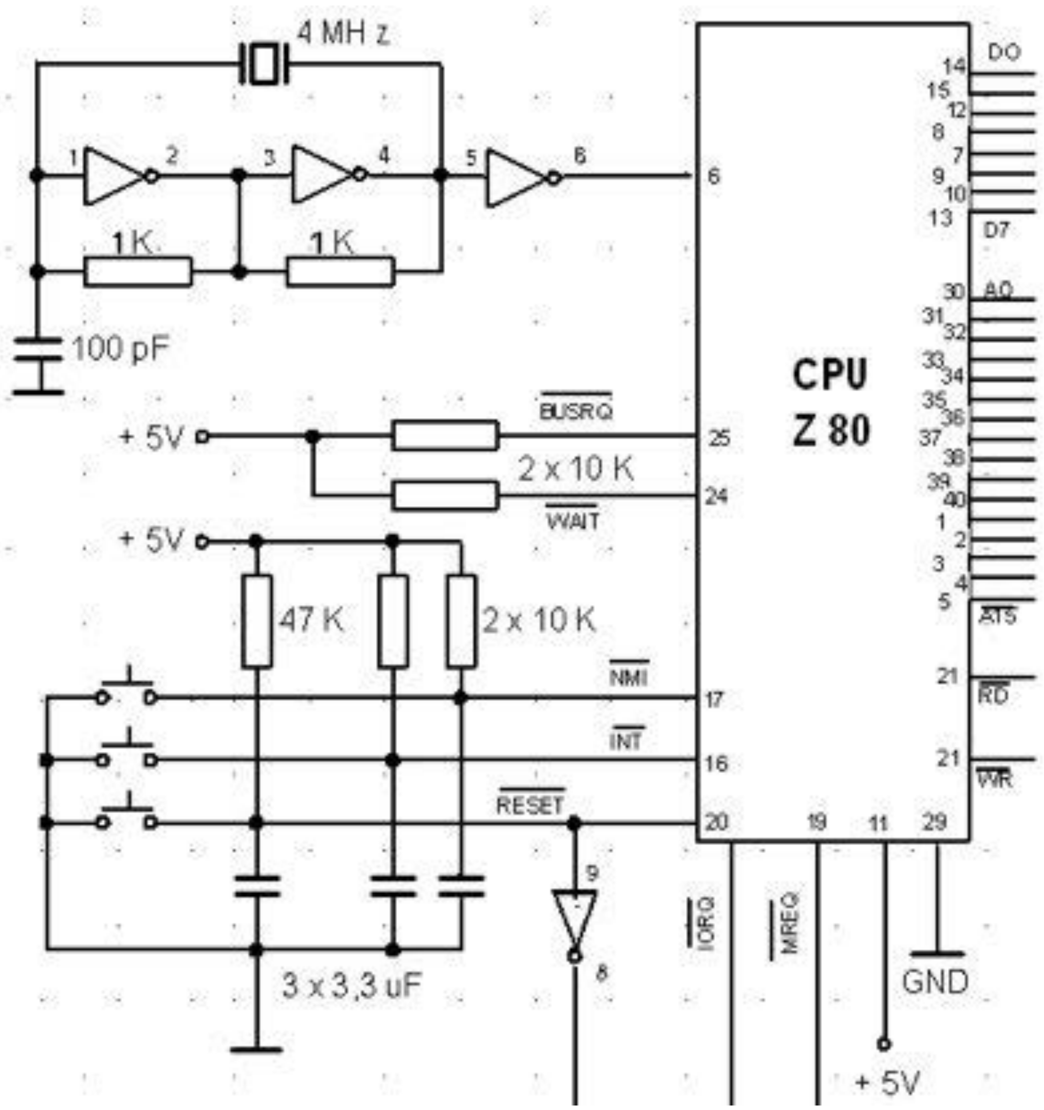
CPU akan mengaktifkan saluran $\overline{WR}$, $\overline{WR}$ = " Low " , selama CPU melaksanakan perintah untuk menulis data ke lokasi penyimpanan data/program (memori) atau register Input output (port).

Untuk saluran - saluran lain pada CPU Z - 80, yang jalannya sinyal sebagai Input, harus diaktifkan sesuai fungsinya dalam sistim.

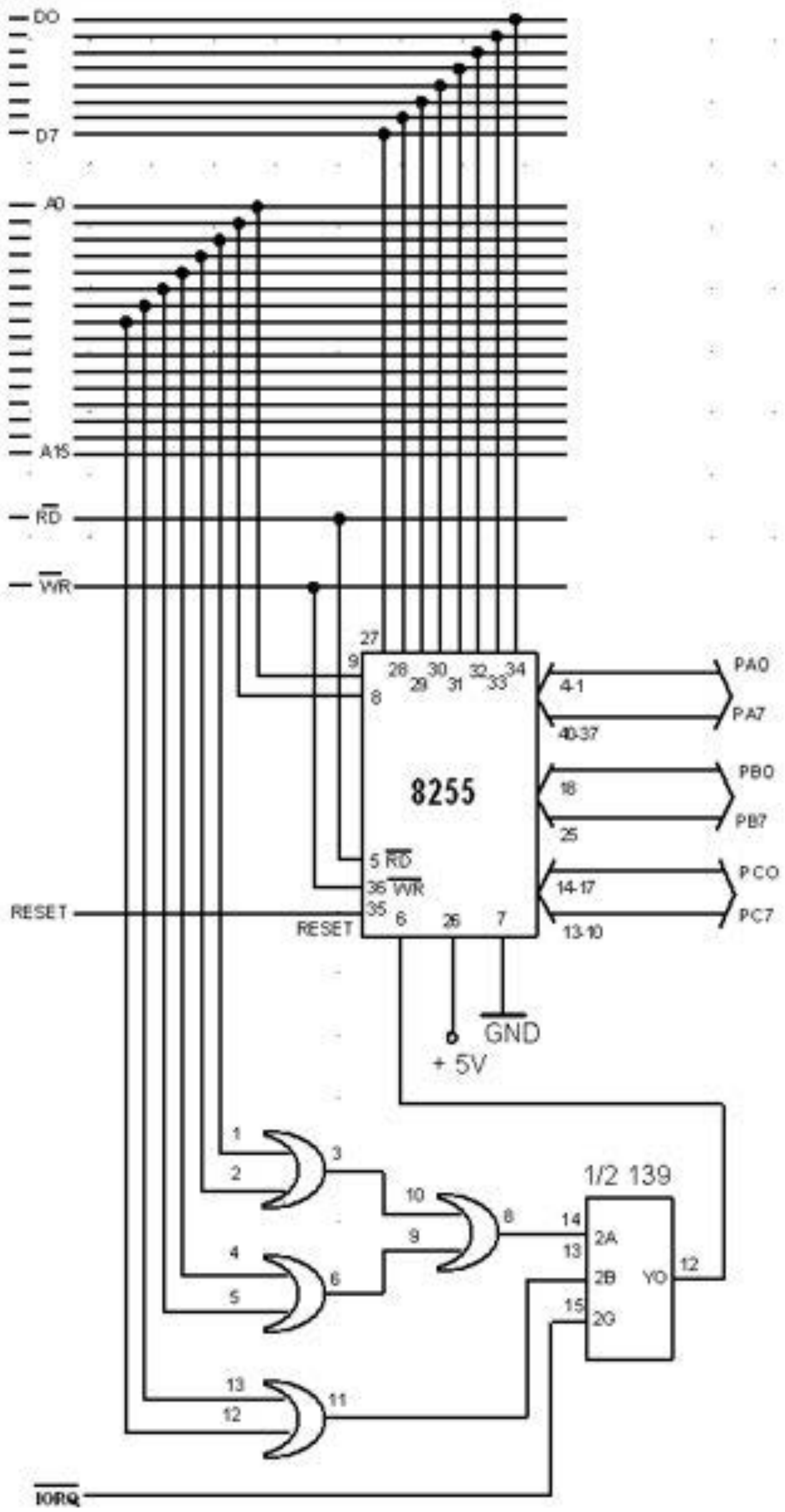


Gambar 3.50. Rangkaia pull up untuk menonaktifkan jalur tidak terpakai.

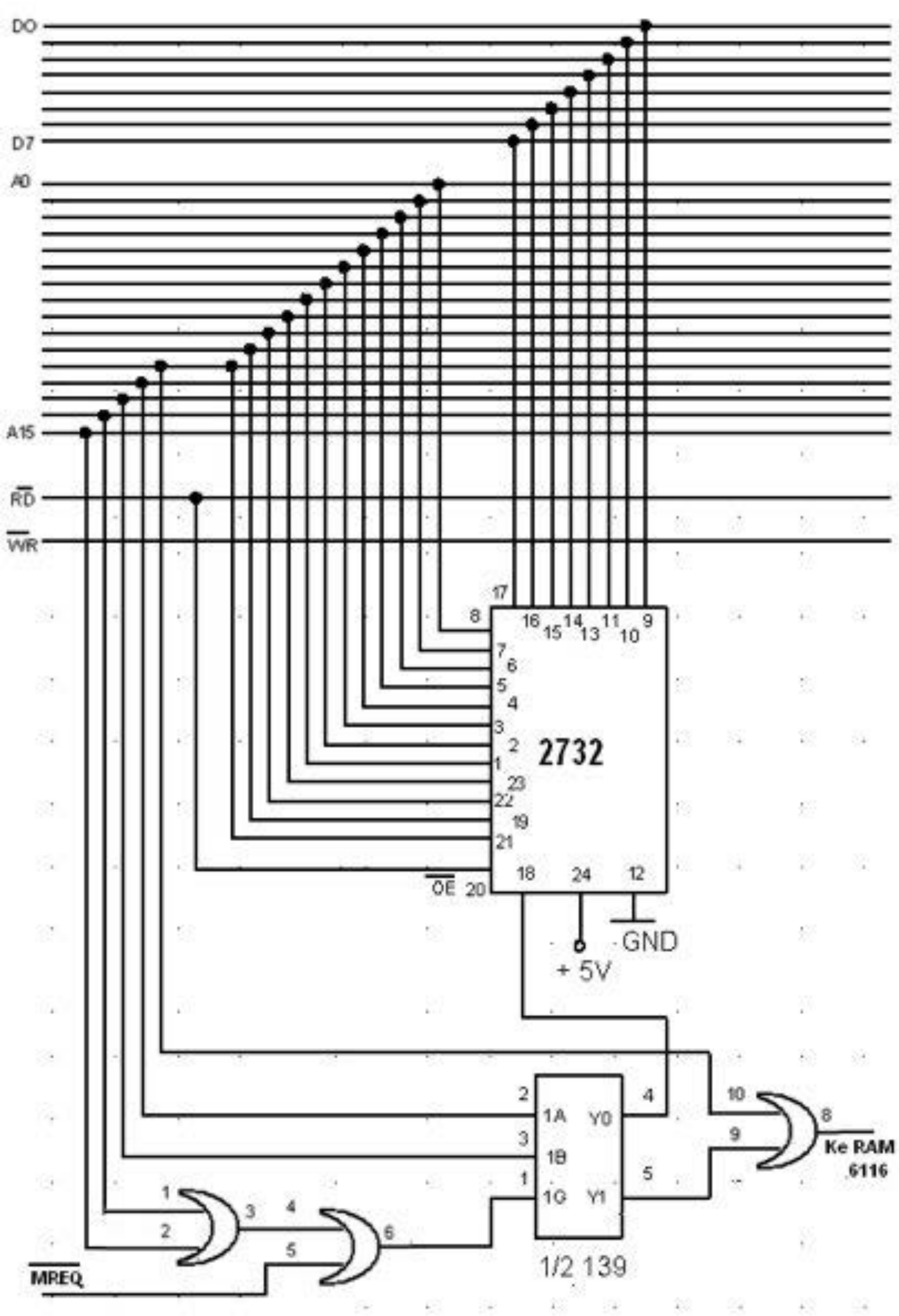
Untuk saluran - saluran lain yang arah jalannya sinyal sebagai output, biarkan tak terhubung kemana - mana.



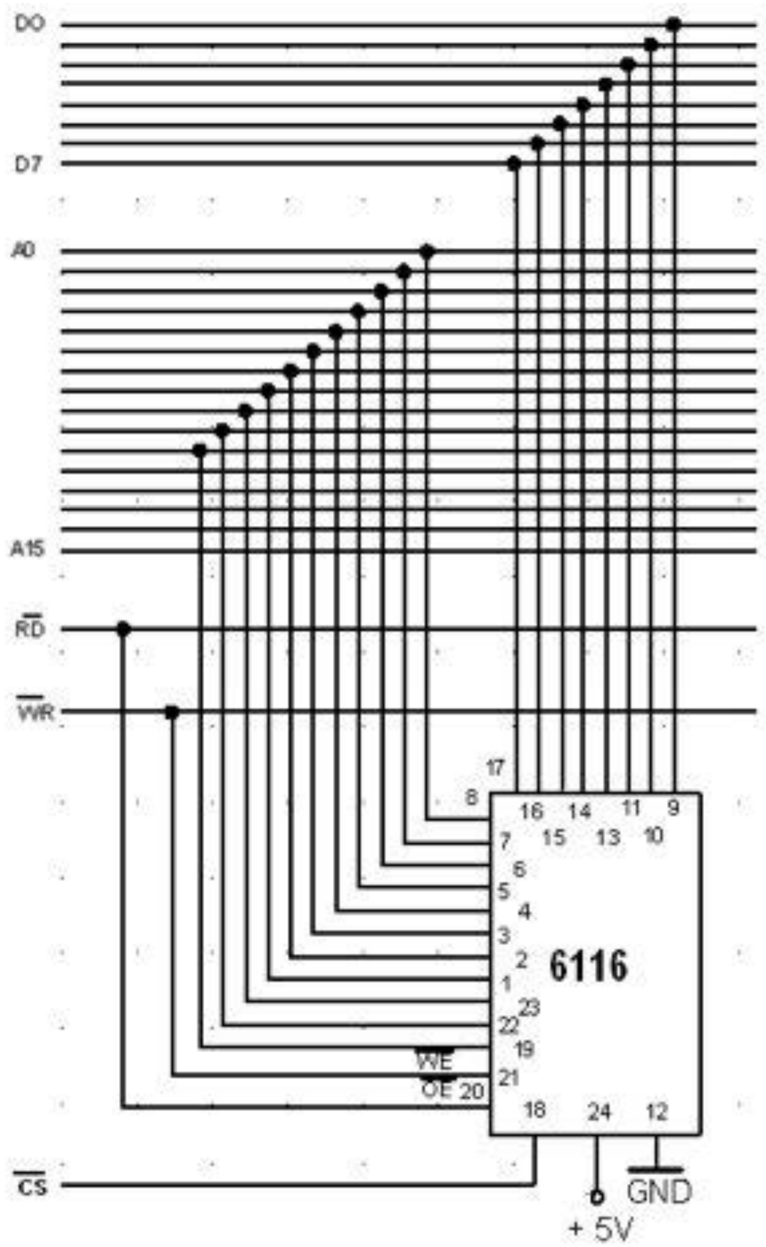
Gambar 3.51. Rangkaian hardware mikroprosesorr Z80



Gambar 3.52. Rangkaian Port I/O PPI 8255 dalam minimal sistem



Gambar 3.53. Rangkaian EPROM 2716 dalam minimal sistem



Gambar 3.54. Rangkaian RAM 6116 dalam minimal sistem

3. 21. Latihan

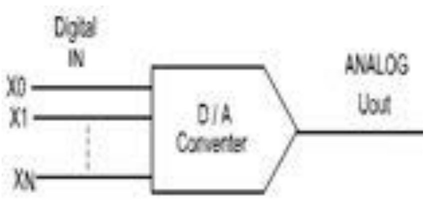
Dengan berpedoman pada lembar data IC, Siswa menjelaskan minimal sistem Z - 80 secara Blok dan menggambarkan layout rangkaian serta membuat rangkaian PCB untuk minimal sistem Z80.

3.22. Pengalih Digital Ke Analog (DAC)

Dalam dunia nyata besaran yang pada umumnya berupa besaran analog, seperti panas, tekanan, kecepatan, arus listrik, tegangan listrik dan sebagainya. Untuk bisa memproses besaran tersebut harus diubah terlebih dahulu ke dalam besarn digital dan apa akhirnya jika diperlukan dapat diubah kembali ke dalam besar analog.

Untuk mendukung konversi tersebut diperlukan Pengalih Digital Ke Analog (DAC), sehingga perlu memahami konsep dasar sistem konversi data analog ke digital dan data digital ke analog. Penjelasan berikut membahas prinsip kerja rangkaian pengalih D/A dan analisa tegangan keluaran pengalih D/A.

Pengalih digital ke analog (digital to analog converter = DAC) merupakan rangkaian yang berfungsi untuk mengubah besaran diskrit atau data-data digital (1 dan 0) menjadi besaran-besaran data analog, berikut merupakan simbol yang digunakan untuk menggambarkan DAC.



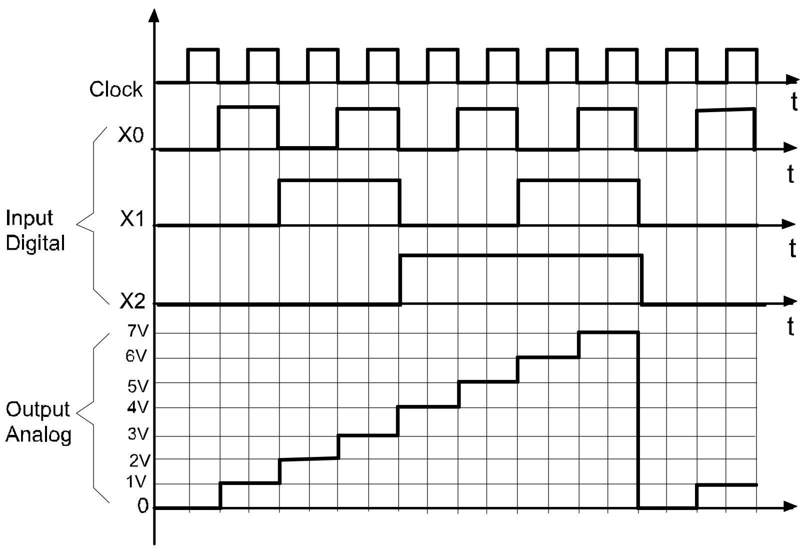
Gambar 3.55. Simbol DAC

Konversi nilai DAC, jika N=2 dan tahapan pertingkat = 1 volt sebagai berikut::

Konversi D/A

INPUT			OUTPUT
X2	X1	X0	VDC

0	0	0	0 Volt
0	0	1	1 Volt
0	1	0	2 Volt
0	1	1	3 Volt
1	0	0	4 Volt
1	0	1	5 Volt
1	1	0	6 Volt
1	1	1	7 Volt



Gambar 3.56. Diagram pulsa hasil konversi D/A

⇒ Gambar 3.55 menunjukkan diagram blok dari pengalih D/A dengan N input masukan digital (X0, X1, ….XN)

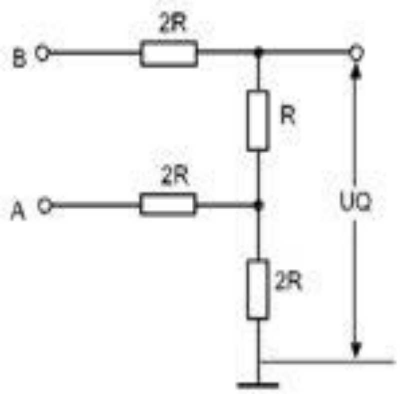
⇒ Variasi input digital X0, X1,dan X2 menunjukkan terjadi perubahan tegangan output analog , seperti yang tampak pada hasil konversi.

⇒ Keluaran dari rangkaian pengalih D/A berbentuk tangga, seperti yang tampak pada gambar 3.56

1. Rancangan rangkaian pengalih D/A

Untuk mempermudah dalam membangun rangkaian pengalih D/A diper-gunakan rangkaian ke resistor dengan teknik R dan 2R, dimana A dan B merupaqkan masukan digital dan R merupakan nilai dasar yang ditentukan.

Contoh :



Rangkaian R dan 2R

Hasil pengukuran UQ

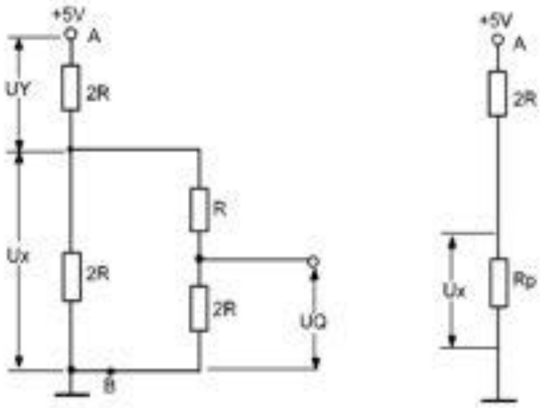
No	INPUT		OUTPUT
	B	A	UQ
1.	0	0 V	0 Volt
2.	0	+ 5V	1,25 Volt
3.	+ 5 V	0 V	2,5 Volt
4.	+ 5 V	+ 5 V	3,75 Volt

Perhitungan U_Q dengan memberikan nilai kombinasi tegangan pada titik A dan titik B, ternyata terjadi hal berikut:

⇒ Perhitungan tegangan output U_Q dilakukan dengan terlebih dahulu merubah rangkaian pengganti.

⇒ Untuk input A = B = 0 ⇒ U_Q = 0 volt

⇒ Untuk input A = + 5V dan B = 0 volt, nilai U_Q adalah:



Rangkaian pengganti RP.

Berdasarkan rangkaian pengganti dan pemberian tegangan pada masukan A sebesar + 5 volt dan pada masukan B = 0 volt, maka U_Q dapat dihitung sebagai berikut:

$$U_Q = \frac{2R}{R+2R} \cdot U_X \Rightarrow U_Q = \frac{2}{3} \cdot 1{,}875\,\text{Volt}$$

$$U_Q = \underline{\underline{1{,}25\,\text{Volt}}}$$

$$Rp = \frac{2R \cdot 3R}{2R + 3R}$$

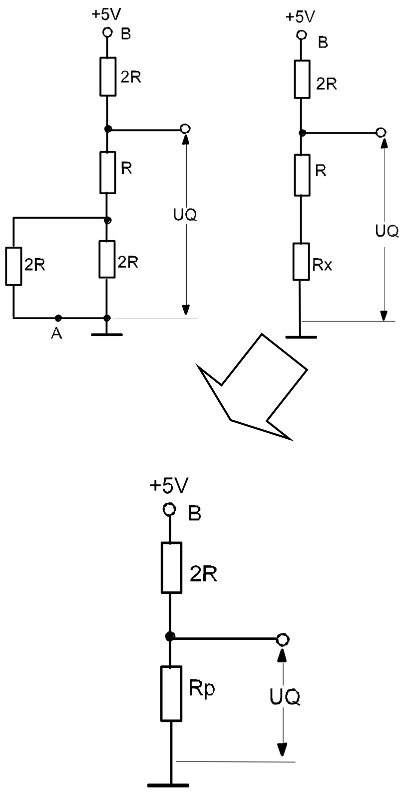
$$= \frac{6}{5}R$$

$$Rp = 1{,}2\,R$$

$$Ux = \frac{1{,}2\,R}{1{,}2\,R + 2R} \cdot 5V$$

$$Ux = 1{,}875\,V$$

Untuk masukan A = 0 volt dan B = + 5 Volt, maka rangkaian pengganti dapat digambarkan sebagai berikut:



$$Rx = \frac{2R.2R}{2R + 2R}$$

$$Rx = \frac{4}{4} R$$

$$Rx = R$$

$$Rp = R + Rx$$

$$Rp = R + R$$

$$Rp = 2R$$

$$U_Q = \frac{Rp}{Rp + 2R} \cdot 5\ V$$

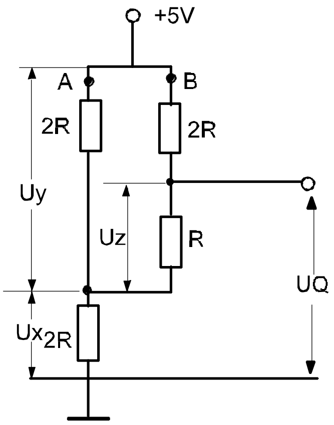
$$U_Q = \frac{2R}{2R + 2R} \cdot 5\ V$$

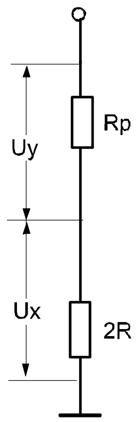
$$U_Q = \frac{2}{4} \cdot 5\ V$$

$$\mathbf{U_Q = 2{,}5\ V}$$

(Rumusan sederhana $UQ = \frac{2}{4} . 5 = 2{,}5$) $U_Q = 2{,}5$ Volt

Untuk masukan A = B = + 5 Volt, maka rangkaian pengganti dapat digambarkan sebagai berikut:





Hasil perhitungan berdasarkan perhitungan rangkaian pengganti, sebagai berikut:

$$Rp = \frac{2R \cdot 3R}{2R + 3R}$$

$$= \frac{6R}{5}$$

$$\mathbf{Rp = \underline{\underline{1{,}2R}}}$$

$$Uy = \frac{1{,}2R}{1{,}2R + 2R} . 5$$

$$\mathbf{Uy = \underline{\underline{1{,}875\ Volt}}}$$

$$Uz = \frac{R}{2R + 3R} \cdot Uy$$

$$Uz = \frac{1}{3} \cdot 1{,}875\ V$$

$$\mathbf{Uz = \underline{\underline{0{,}625\ V}}}$$

$$Ux = UB - Uy$$

$$Ux = 5V - 1{,}875\ V$$

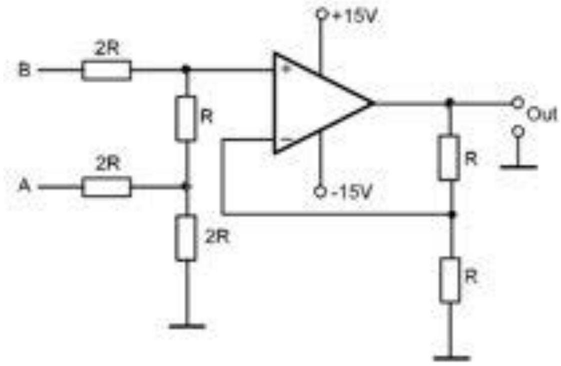
$$\mathbf{Ux = \underline{\underline{3{,}125\ V}}}$$

$$U_Q = Uz + Ux$$

$$U_Q = 0{,}625\ V + 3{,}125\ V$$

$$\mathbf{U_Q = \underline{\underline{3{,}75\ V}}}$$

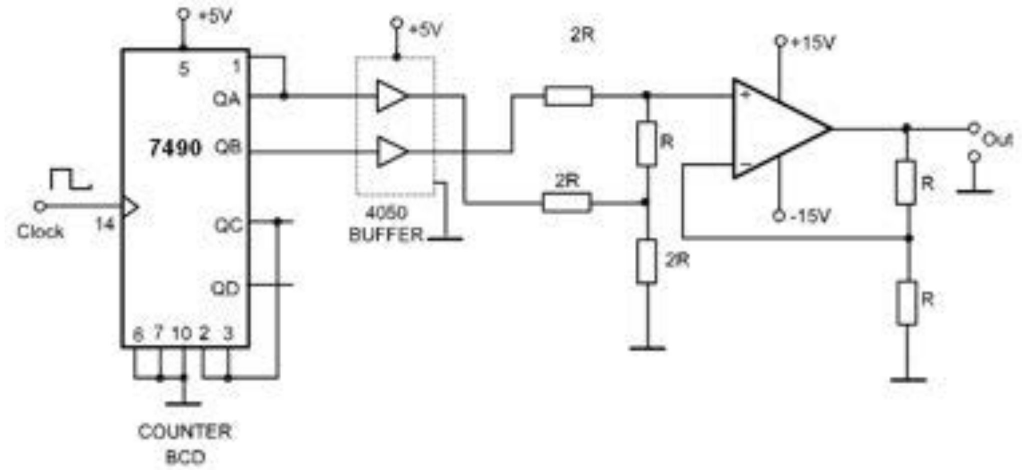
Dengan diperolehnya rangkaian R dan 2R dan dari perhitungan diperoleh hasil perhitungan UQ, dengan menambahkan rangkaian penguat maka didapatkan rangkaian pengalih D/A.



Gambar 3.57. Rangkaian pengalih Digital ke Analog

Jika rangkaian pengalih Digital ke Analog tersebut pada masukan A dan masukan B diumpankan pada sebuah penghitung BCD, maka dengan diberikannya clock penghitung akan mulai menghitung mulai dari 0, 1, 2, 3 dan kembali lagi ke 0.

Perubahan keluaran penghitung (*counter*) akan memberikan kombinasi logika pada masukan A dan B, sehingga diperoleh hasil konversi pada Uout.



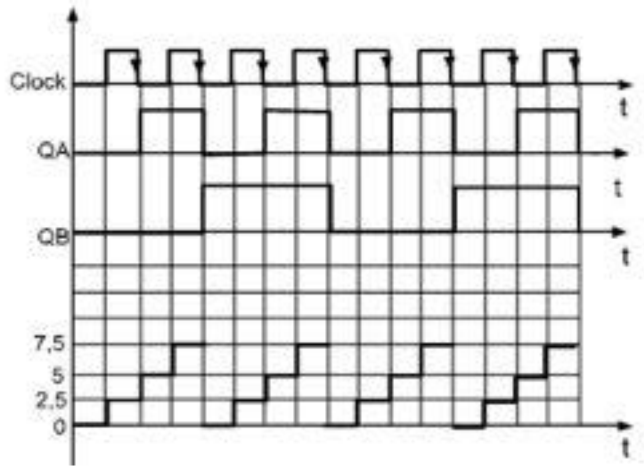
Gambar 3.58. Rangkaian pengalih Digital ke Analog dengan *Counter*

⇒ U_{out} diperoleh dari perhitungan U_Q dikalikan dengan penguatan rangkaian OP-AMP non inverting yaitu Av = 2x, sehingga diperoleh hasil sebagai berikut:

	INPUT		OUTPUT
Clock	Q_B	Q_A	U_{out}

0.	0	0	0 Volt
1.	0	1	2,5 Volt
2.	1	0	5 Volt
3.	1	1	7,5 Volt
4.	0	0	0 Volt

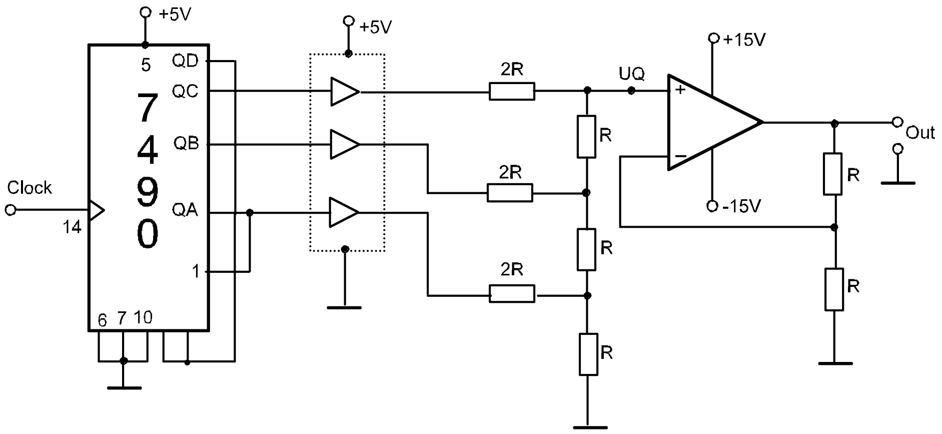
Dari hasil penguatan dapat di gambarkan secara diagram sebagai berikut:



Gambar 3.59. Diagram pulsa pengalih D/A dengan *counter*.

Latihan

1. Hitunglah tegangan keluaran dari rangkaian pengalih digital ke anlaog di bawah ini ! (gambar yang termudah).



Jawaban

Out 7490			**Out OP-AMP**
Q_C	**Q_B**	**Q_A**	**U_{out}**
0	0	0	0 Volt
0	0	1	1,25 Volt
0	1	0	2,5 Volt
0	1	1	3,75 Volt
1	0	0	5 Volt
1	0	1	6,25 Volt
1	1	0	7,5 Volt

$$U_Q = \frac{\text{Nilai biner}}{2^3}.5$$

$$U_{out} = U_Q \cdot 2$$



1	1	1	8,75 Volt

a) Untuk $Q_A = Q_B = Q_C = 0$, $U_{out} = 0$ Volt

b) Untuk $Q_A = 1$, $Q_B = Q_C =$ "0"

$$U_Q = \frac{\text{Nilai akhir}}{2^3}.5$$

$$U_Q = \frac{1}{8}.5 = 0{,}625 \text{ Volt}$$

$$U_{out} = 0{,}625 \;.\; 2 = 1{,}25 \text{ Volt}$$

c) Untuk $Q_A = Q_C =$ "0" ,$Q_B =$ "1

$$U_Q = \frac{2}{8}.5$$

$U_Q = 1{,}25$ Volt

$U_{out} = 1{,}25.2 = 2{,}5$ Volt

d) Untuk $Q_A = Q_B =$ "1" ,$Q_C =$ "0

$$U_Q = \frac{3}{8}.5$$

$U_Q = 1{,}875$ Volt

$U_{out} = 1{,}875 \;.\; 2 = 3{,}75$ Volt

e) Untuk $Q_A = Q_B =$ "0" ,$Q_C =$ "0

$$U_Q = \frac{4}{8}..5 = 2{,}5 \text{ Volt}$$

$U_{out} = 2{,}5 \;.\; 2 = 5$ Volt

f). Untuk $Q_A = Q_C =$ "1" ,$Q_B =$ "0"

$$U_Q = \frac{5}{8}.5 = 3{,}125 \text{ Volt}$$

$U_{out} = 3{,}125 \;.\; 2 = 6{,}25$ Volt

g). Untuk $Q_B = Q_C =$ "1" ,$Q_A =$ "0"

$$U_Q = \frac{6}{8} \;.\; 5 = 3{,}75 \text{ Volt}$$

$U_{out} = 3{,}75 \;.\; 2 = 7{,}5$ Volt

h). Untuk $Q_A = Q_B = Q_C =$ "1"

$$U_Q = \frac{7}{8} \;.\; 5 = 4{,}375 \text{ Volt}$$

$U_{out} = 4{,}375 \;.\; 2 = 8{,}75$ Volt

3.23. Pengalih Analog Ke Digital (ADC)

Pengalih analog ke digital yang dikenal juga dengan ADC, yang merupakan rangkaian elektronik yang mampu mengubah besaran analog menjadi besaran digital.

Untuk dapat memahami ADC dengan benar dapat dilakukan dengan mengenali dan memahami hal berikut:

- ⇒ ciri-ciri pengalih analoh ke digital serempak
- ⇒ prinsip rangkaian pengalih analog ke digital serempak
- ⇒ keuntungan pengalih analog ke digital serempak
- ⇒ kerugian pengalih analog ke digital serempak.

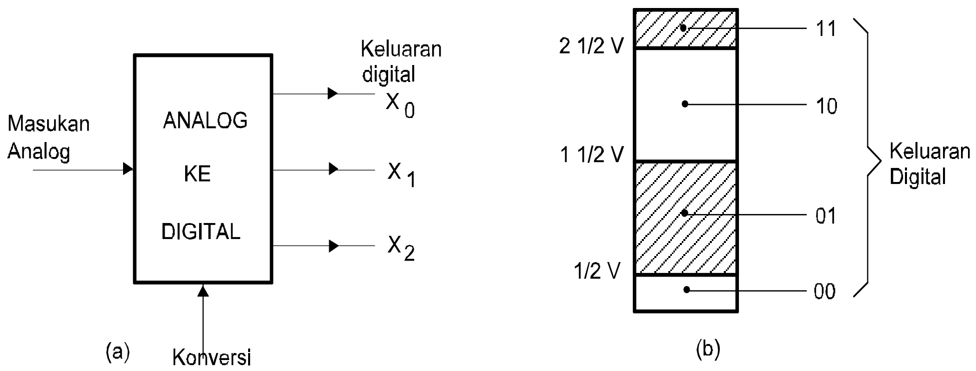
1. Ciri-Ciri Pengalih Analog Ke Digital Serempak

Sebuah ADC memiliki bagian-bagian:

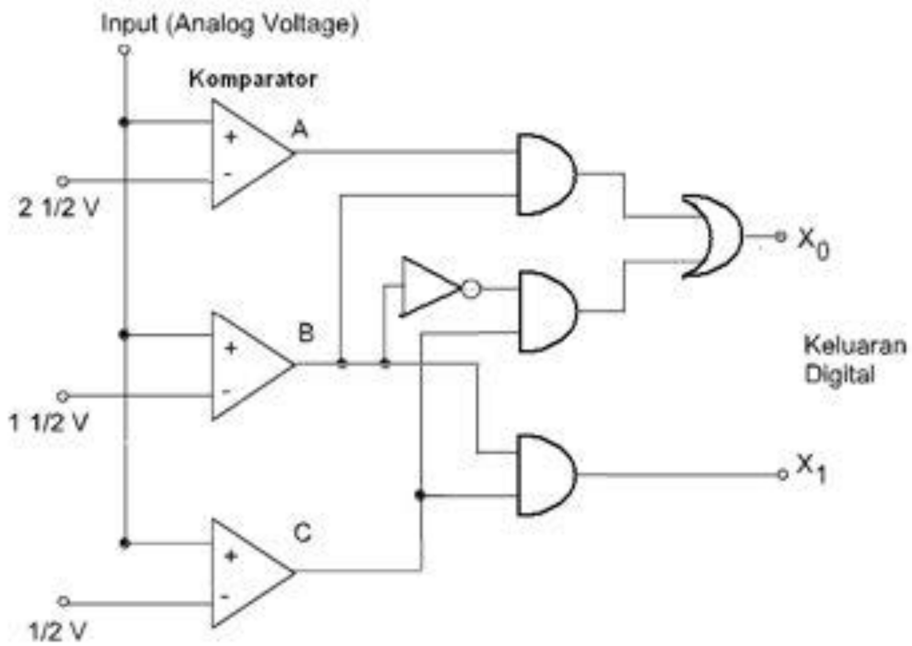
- a Pembanding / voltage comparator
- b Penghitung / counter
- c Pengubah digital ke analog (D/A counter), ini bila menggunakan prinsip kerja dari digital ke analog

2. Prinsip Rangkaian Pengalih Analog Ke Digital Serempak

Konversi besaran analog ke besaran digital dapat dilakukan dengan gambaran prinsip berikut:

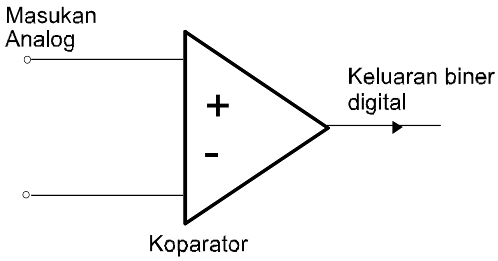


Gambar 3.60. Konversi besaran analog ke besaran digital



Gambar 3.61. Rangkaian pengalih analog ke digital

Catatan :



jika masukan (+) poistip terhadap masukan (-) keluaran digital ialah 1.

Jika masukan (=) positif terhadap masukan (-) keluaran digital ialah 0

Analisis Kerja:

a. Jika masukan 0 hingga 1/2 V, maka ketiga titik A, B dan C akan 0, X_0 dan X_1 juga 0.

b. Jika masukan lebih dari 1/2 V hingga 1 1/2 V, maka titik A dan B akan 0, tetapi C menjadi 1, dan keluaran X_0 menjadi 1, X_1 menjadi 0 .

c. Jika masukan lebih dari 1 1/2 V hingga 2 1/2 V, titik B dan C akan jadi 1 dan A adalah 0 atau $X_0 = 0$ dan $X_1 = 1$.

d. Jika masukan lebih dari 2 1/2 V, maka titik A, B dan C berharga 1 dan keluaran $X_0 = 1$, $X_1 = 1$.

3. Keuntungan Pengalih Analog Ke Digital Serempak

Input analog voltage serempak masuk baik tegangan rendah sampai tegangan maksimum langsung dibandingkan dengan tegangan referensi yang ada. Kecepatan proses tinggi.

4. Kerugian Pengalih Analog Ke Digital Serempak

Dipakai dengan jumlah bit yang banyak kurang efisien dan lebih rumit , sulit, karena komparator meningkat (2^n - 1). n = jumlah bit.

Latihan

1. Sebutkan ciri-ciri pengalih analog ke digital serempak !
2. Buatlah rancangan rangkaian pengalih analog ke digital serempak 3 bit !

Jawaban

1. Pembanding / comparator voltage, Penghitung / pembaca
2. Rangkaian pengalih A/D serempak

